

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Центральноукраїнський національний технічний університет

Кафедра кібербезпеки та програмного забезпечення

Думенко Олег Андрійович

**Програмне забезпечення ігрового лаунчера комп'ютерної гри
World of Warcraft**

Спеціальність: 123 «Комп'ютерна інженерія»

Освітній ступінь: бакалавр

Науковий керівник:

Савеленко Олена Константинівна,

викладач

_____ (підпис)

_____ (дата)

ДОПУЩЕНО ДО ЗАХИСТУ

Завідувач кафедри

(_____) **Смірнов О.А.**
(підпис) ПІБ

_____ 20 р.

Міністерство освіти і науки України
Центральноукраїнський національний технічний університет
Факультет Механіко-технологічний
Кафедра Кібербезпеки та програмного забезпечення
Освітньо-кваліфікаційний рівень бакалавр
спеціальність 123 “Комп’ютерна інженерія”

ЗАТВЕРДЖУЮ
Завідувач кафедри
д.т.н., проф.
_____ **О.А.Смірно**
В
“ _____ ”
_____ 20__ року

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТУ

Думенку Олегу Андрійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи Програмне забезпечення ігрового лаунчера комп’ютерної гри World of Warcraft

керівник роботи Савеленко Олена Константинівна, викладач

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затвержені наказом закладу вищої освіти від “ _____ ” 20__ року № _____

2. Строк подання студентом роботи 22.05.2021 р.

3. Мета та завдання випускної кваліфікаційної роботи

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Призначення та область використання.

2. Перегляд аналогічних існуючих систем.

3. Опис і обґрунтування проектних рішень.

4. Реалізація роботи. Розрахунки та експериментальні дані, підтверджують вірність проектних та програмних рішень.

5. Впровадження системи в промислову експлуатацію.

6. Висновки.

5. Перелік графічного матеріалу (з точним зазначенням обов’язкових креслень)

Структурна схема системи 1 аркуш

Функціональна схема системи 1 аркуш

Діаграма процесів 1 аркуш

Блок-схема алгоритму роботи додатку 3 аркуши

6. Дата видачі завдання « » 20 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної бакалаврської роботи	Строк виконання етапів кваліфікаційної бакалаврської роботи	Примітки
1.	Аналіз існуючих систем керування	10.03.2021 р.	
2.	Постановка задачі, оформлення ТЗ	15.03.2021 р.	
3.	Розробка моделі компонента	20.03.2021 р.	
4.	Розробка структур даних	25.03.2021 р.	
5.	Розробка алгоритмів зв'язку та відображення	30.03.2021 р.	
6.	Програмування алгоритмів	10.04.2021 р.	
7.	Оформлення ПЗ	17.05.2021 р.	
8.	Попередній захист роботи	22.05.2021 р.	

Студент _____ (підпис) _____ (прізвище та ініціали)

Керівник роботи _____ (підпис) _____ (прізвище та ініціали)

АНОТАЦІЯ

Думенко О.А. Програмне забезпечення ігрового лаунчера комп'ютерної гри World of Warcraft. 123 Комп'ютерна інженерія. Центральноукраїнський національний технічний університет. Кропивницький. 2021.

В даній кваліфікаційній бакалаврській роботі розроблено програмне забезпечення, яке призначено для комп'ютерної гри World of Warcraft.

Метою розробки є програмне забезпечення лаунчер комп'ютерної гри World of Warcraft.

Результат роботи – програмна для запуску ігрового лаунчера для комп'ютерної гри.

В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

Розроблено зручний інтерфейс користувача. Наведені інструкції по роботі з програмними засобами.

Програма може використовуватися на ПЕОМ архітектури IBM PC з ОС Windows XP/7/8/10.

Програму розроблено в середовищі Microsoft Visual Studio на мові C#.

Ключові слова: Ігровий лаунчер, рекомендаційні системи, автоматичність додатку.

ABSTRACT

Dumenko O.A. Game launcher software for World of Warcraft computer game. 123 Computer Engineering. Central Ukrainian National Technical University. Kropyvnytskyi 2021

In this bachelor qualification developed software that is designed for the computer game World of Warcraft.

The purpose of the development is the game launcher software for World of Warcraft computer game.

The result is a software game launcher for a computer game.

In the process of working on a software model an analysis of existing hardware and software was performed. All components of the software developed are fully described.

User-friendly user interface is developed. These are instructions for working with software.

The program can be used on an IBM PC running Windows XP/7/8/10.

The program was developed in Microsoft Visual Studio in C#.

Keywords: Game launcher, recommendation systems, application automation.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ	3
ВСТУП.....	4
1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ.....	6
1.1 Призначення системи.....	6
1.2 Область застосування.....	9
2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ	11
2.1 Огляд існуючих систем, технологій, архітектур та програмних рішень по профілю теми дипломної роботи	11
2.2 Обґрунтування вибору засобів для побудови системи та мови програмування.....	18
3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ	23
3.1 Опис функціонування системи.....	23
3.2 Розробка структурної схеми	33
3.3 Розробка функціональної схеми	34
3.4 Розробка діаграми процесів	36
4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ ТА ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ І ПРОГРАМНИХ РІШЕНЬ	38
4.1 Розробка блок-схем та опис алгоритмів функціонування системи	38
4.2 Захист розробленого програмного забезпечення	45
5 ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ.....	49
5.1 Експлуатація і індивідуальний підхід до промислової експлуатації.....	49
5.2 Інструкція користувача	50
5.3 Інструкція інженера.....	51
5.4 Методи та обґрунтування тестування програмного продукту.....	59

КБР-123.21.0051.00.00.ПЗ

Вим.	Арк.	№ докум.	Підп.	Дата				
Розроб.		Думенко О.А.			Програмне забезпечення ігрового лаунчера комп'ютерної гри <i>World of Warcraft</i>	Лім.	Аркуш	Аркушів
Перев.		Савеленко О.К.				Б	1	69
Н.контр.		Гермак В.С.			ЦНТУ КІ-19 2ск			
Затв.		Смірнов О.А.						

7 ОСНОВНІ ВИСНОВКИ..... 63

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ..... 65

Кафедра _ КБПЗ _ 2021 рік

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		2

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

- Рис. – Рисунок
- SSD - комп'ютерний незалежний немаханічний пристрій
- Див. – Дивитись
- Лаунчер – програмна оболонка
- XML – розширювана мова розмітки
- C# - об'єктно-орієнтована мова програмування
- .NET - платформа від Microsoft
- CLR
- EXE - розширення виконуваних файлів
- SSD - комп'ютерний незалежний немаханічний пристрій
- Node.js - програмна платформа
- DLL - динамічне підключення бібліотеки
- HASP - апаратно-програмна система захисту програм
- CLR - загальномовне виконуюче середовища
- ASCII - назва таблиці (кодування, набору)
- ini - це файл конфігурації, який містить дані налаштувань

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		3

- Побудувати блок-схему алгоритму основної програми та підпрограм та виконати їх опис;

- Розробити програмний продукт та виконати його тестування;

- Розробити інструкцію користувача та адміністратора.

Розроблений програмний продукт повинен мати функції:

- Кнопки переходу на веб-ресурси;

- Автоматична авторизація на сервер;

- Чистка кеш-пам'яті гри;

- Оновлення і скачування патчів гри;

- Перевірка версій та автоматичне оновлення клієнту;

- Автоматичне оновлення програми;

- Інформація під час оновлення або скачування патчів (Показує % скачування, швидкість скачування поновлення);

- Вивід новин

- Заборона запуску інструментарію повторно (Поки запущений клієнт WoW);

Цінність програмного продукту полягає в тому, що розроблений лаунчер має низку переваг над звичайними лаунчерами. Однією з переваг реалізації автоматичного оновлення клієнту гри. Окрім того, лаунчер дає змогу зберегти час користувача, а його, як відомо, замінити і повернути неможливо.

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		5

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Даний додаток розроблявся для особистого ігрового проекту гри World of Warcraft, на якому я є куратором і розробником проекту, який на момент написання і здачі дипломного проекту запущений і працює на базі даного лаунчер. Яким користується понад 14 000 чоловік.

Додаток допомагає гравцям швидко, а головне дуже зручно оновлювати файли ігрового сервера, щоб без всяких зусиль і зайвих дій потрапити на сервер.

Основне призначення цієї програми - для користування гравцями ігрового проекту.

У перекладі з англійської мови слово «Launcher» означає «той, що допомагає запуску». Це спеціалізована комп'ютерна програма, призначена для полегшення запуску та управління іншими програмами, в моєму випадку програма допомагає запуску гри World of Warcraft.

Завантажувальний лаунчер для Windows допомагає при виконанні завантаження: розподіляє ресурси системи між додатками, завдяки чому загальний час завантаження скорочується. Запуск лаунчер як правило, відбувається при виконанні програм автозавантаження, в цьому ж вікні можна відключити і деінсталювати скачані файли.

Найсудоріше і єдине правило побудови систем програмного забезпечення (ПЗ) – вирішити точно, що ж будувати. Ніяка інша частина концептуальної роботи не є такої важкої, як з'ясування деталей технічних вимог, у тому числі й взаємодія з людьми, з механізмами й з іншими системами ПО. Ніяка інша частина роботи так не псує результат, якщо вона виконана погано. Помилки ніякого іншого етапу роботи не виправляються так важко (кінець цитати). Наука добування й формалізації якісних (іноді говорять «гарних», «правильних») вимог

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		6

носить багато в чому емпіричний характер. Однак, у практиці розробки програмних систем нагромадилися певні подання про те, якими властивостями повинні володіти вимоги до програмної системи. Це:

- Повнота;
- Якість;
- Коректність;
- Погодженість;
- Необхідність;
- Корисність при експлуатації.

Більшість цих властивостей виконана в моєму дипломному проекті до кінця. Додадок має повний набір функцій для гри на ігровому сервері, якісну та комфортну зовнішню та функціональну оболонку.

В дизайні буде передбачено наявність окремих стилів для певних типів інформації (наприклад, для інформації версії додатку або слайдів). Дизайн лаунчеру - дуже важливий елемент впливу на ігрову інтернет-аудиторію. При створенні і розробці додатку, буде звернено увагу на комфортне і зручне сприйняття інформаційного змісту додатку. Текст не повинен зливатися з фоном кнопок та налаштувань, інакше він буде не читабельним. А це негативним чином позначиться на популярності ігрового проекту.

Також великі тексти, для забезпечення зручного читання, буде розділено на глави і підзаголовки (в слайді додатку).

Основним завданням дизайну є:

- Ергономіка - легкість у використанні об'єкту
- Інтерфейс користувача - психологічні особливості оператора, працюючого з об'єктом дизайну
- Останні тенденції - застосування нових технологій в дизайні для спрощення його інших завдань.

Функції, які виконує дизайн додатку:

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		7

- Залучення увага користувачів додатку за допомогою кольорових рішень, шрифтового оформлення, графічних елементів, анімації;
- Створення унікального зовнішнього вигляду, який відрізняє компанію від конкурентів;
- Дає можливість ефективно розмістити вичерпне інформаційне наповнення, яке буде доступним і читабельним.

Згідно з технічним завданням на кваліфікаційну бакалаврську роботу, реалізації підлягає програмне забезпечення, яке призначено для комп'ютерної гри World of Warcraft.

В процесі розробки кваліфікаційної бакалаврської роботи необхідно виконати наступний обсяг роботи:

а) провести аналіз існуючих систем-аналогів для виявлення їх позитивних і негативних якостей. Результати аналізу врахувати в подальших розробках;

б) вибрати та обґрунтувати методику побудови системи контролю роботи технологічного обладнання на виробництві в автоматизованому режимі. Розробити функціональну та структурну схеми системи;

в) розробити програмне забезпечення системи, що дозволить реалізувати поставлену технічним завданням задачу. Побудувати блок-схеми алгоритмів програми та підпрограми;

г) організувати інтерфейс користувача з метою формування та виводу на екран ЕОМ повідомлень про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання;

д) розробити рекомендації по організаційних та методичних заходах, які забезпечать впровадження системи в промислову експлуатацію та її подальшу успішну експлуатацію;

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

ж) розробити заходи по охороні праці при впровадженні та експлуатації системи, а також розробити заходи з цивільного захисту;

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		8

Вирішальні фактори успішності проекту:

- як швидко користувачі отримують необхідну інформацію;
- в якому вигляді вона представлена;
- її простота і доступність для будь-якого споживача.

Від цього залежить рішення прочитати текст до кінця, заходити на ігровий сервер, перейти на сайт, виконати іншу очікувану дію. Наскільки повно компанії/власників особистого проекту заявлять про себе, знайдуть можливість залучити й утримати відвідувачів, залежить успіх і розкрутка проекту, вибір способів просування додатку.

Частина відповідальності за інформацію залежить від якості текстів та оформлення, які написали в додатку.

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		10

2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

2.1 Огляд існуючих систем, технологій, архітектур, програмних рішень за профілем теми кваліфікаційної бакалаврської роботи

Sirus.su – Ігровий проект World of Warcraft

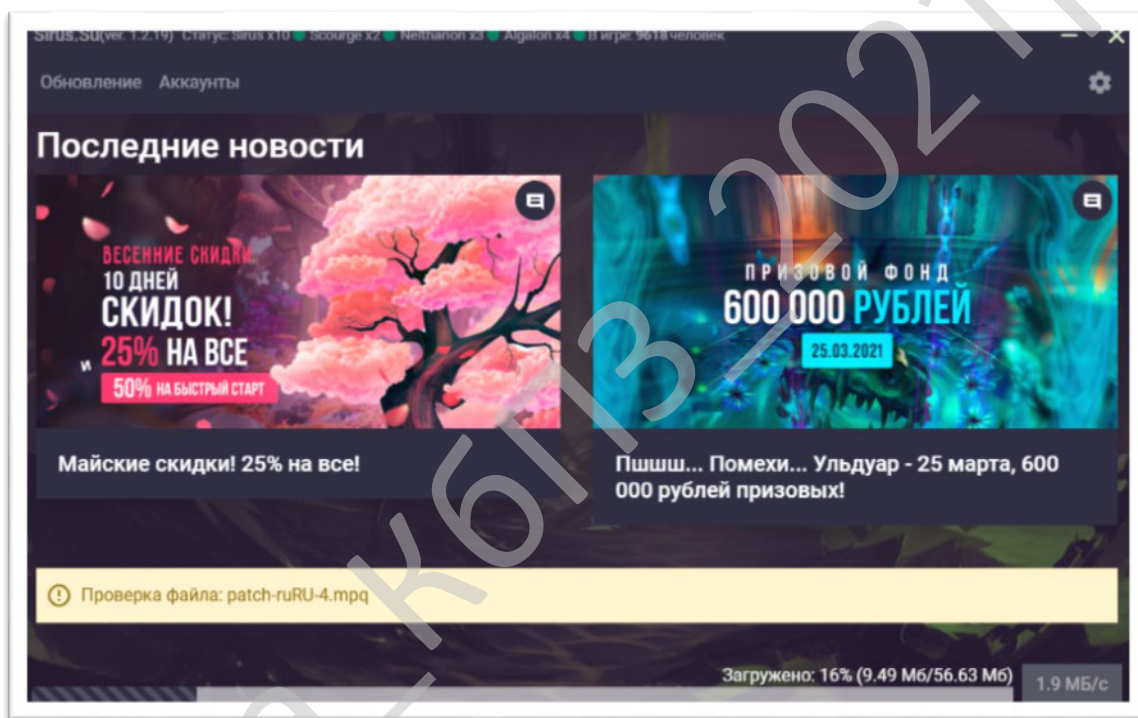


Рисунок 2.1 – Лаунчер проекту Sirus.su

У верхній частині лаунчер присутній індикатор стану сервера, в залежності від кольору він відображає, чи знаходиться сервер, в даний момент, онлайн або оффлайн.

Нижче є дві кнопки "Оновлення" і "Акаунти". Кнопка "Оновлення" потрібна для активації поновлення клієнта або гри, друга кнопка - "Акаунти" призначена для додавання профілів і перемикання між ними.

У самому правому куті є кнопка у вигляді шестерінки, вона відкриває меню налаштувань.

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		11

Найбільше простору тут займає внутрішнє вікно з останніми новинами, яке відображає найсвіжіші новини, які дозволяють заощадити час і швидко розібратися що нового відбулося з грою.

У самому низу є кнопка "перевірити клієнт" вона дозволяє запустити перевірку файлів лаунчер і додати відсутні файли, якщо вони були видалені.

Кнопка "проблеми з підключенням" перенаправляє користувача на форум, де він може описати адміністратору проблему і вирішити її. Права кнопка "в гру" запускає саму гру.

При натисканні на кнопку "Налаштування" відкривається вікно з трьома вкладками - "Налаштування лаунчеру", "Налаштування клієнта" і "Експертні настройки". У вкладці "Налаштування лаунчеру" можна вказати розташовані самі гри, включити або відключити деякі функції лаунчеру, дізнатися дату останнього оновлення клієнта, відкрити папку з одними даними, дізнатися глобальний ідентифікатор і кількість частково завантажених файлів.

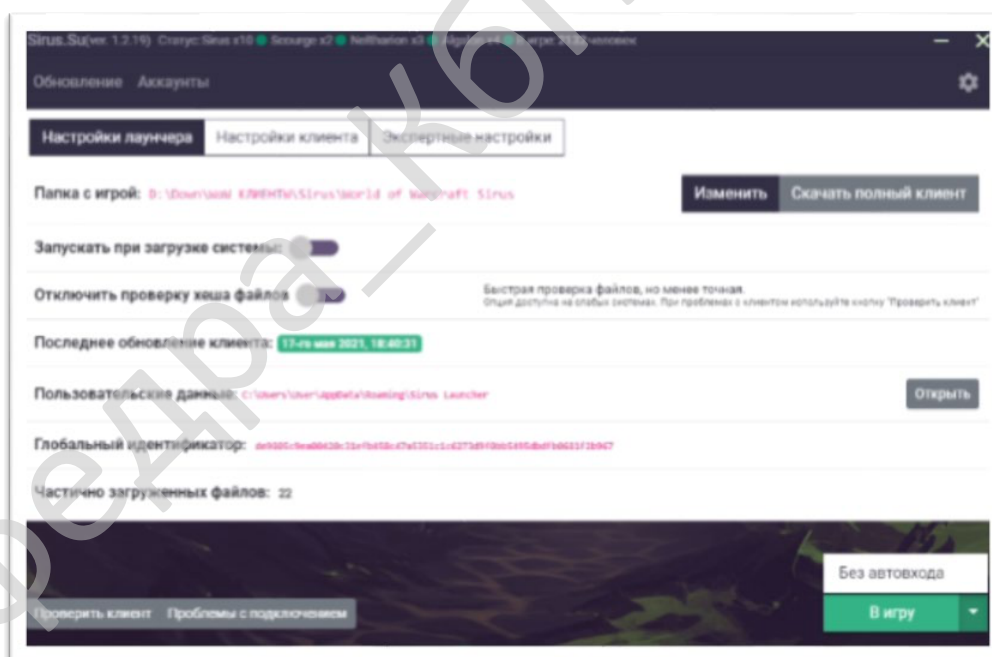


Рисунок 2.2 – Вкладка налаштування лаунчеру

У вкладці "Налаштування клієнта" можна вибрати режим вікна - повноекранний або віконний і вказати розширення екрану.

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		12

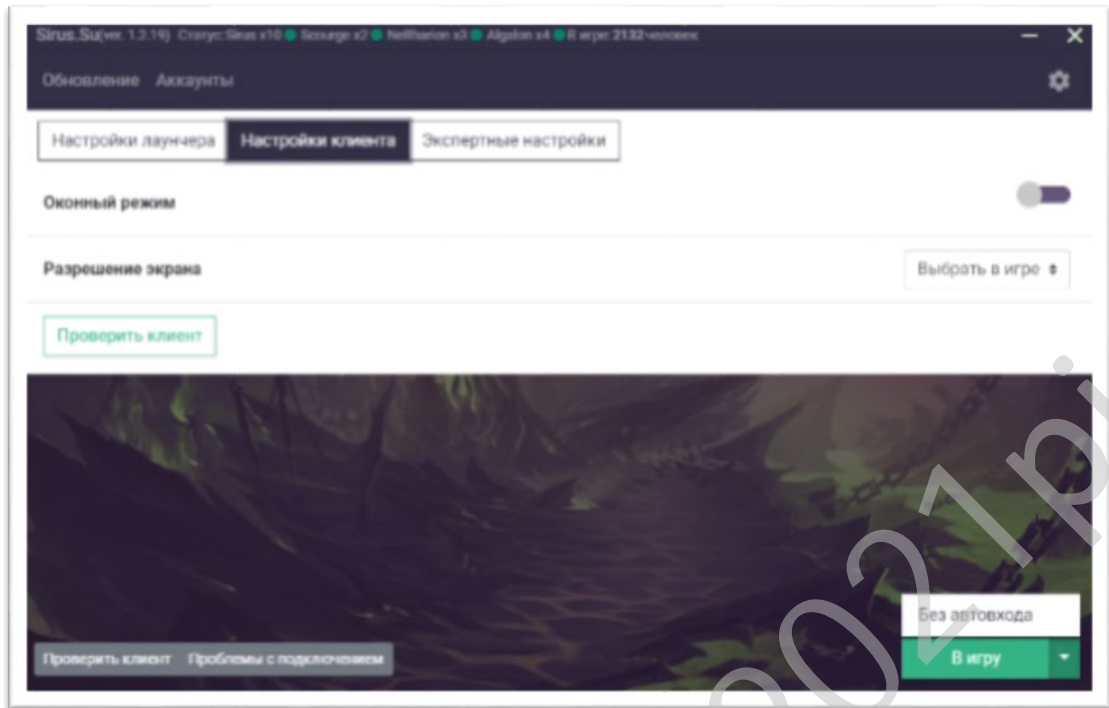


Рисунок 2.3 – Вкладка налаштування клієнта

У вкладці "Експертні налаштування" є можливість вказати особливі параметри запуску гри, на вибір користувача.

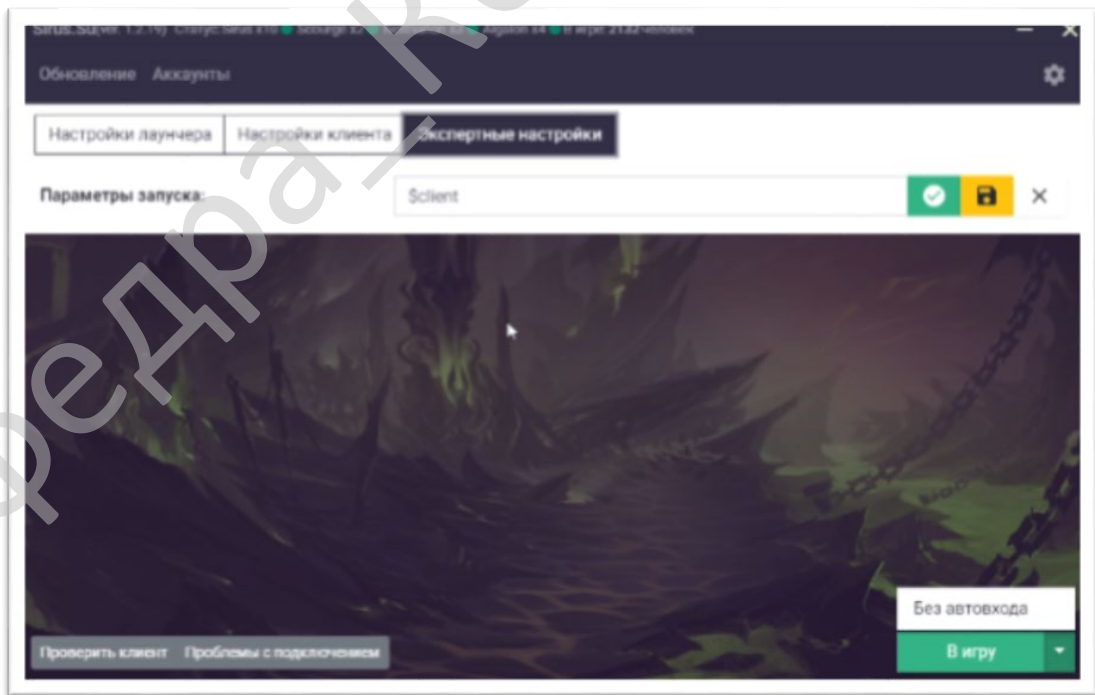


Рисунок 2.4 – Вкладка експертні налаштування

Вим.	Арк.	№ докум.	Підпис	Дата

КБР-123.21.0051.00.00.ПЗ

Арк.

13

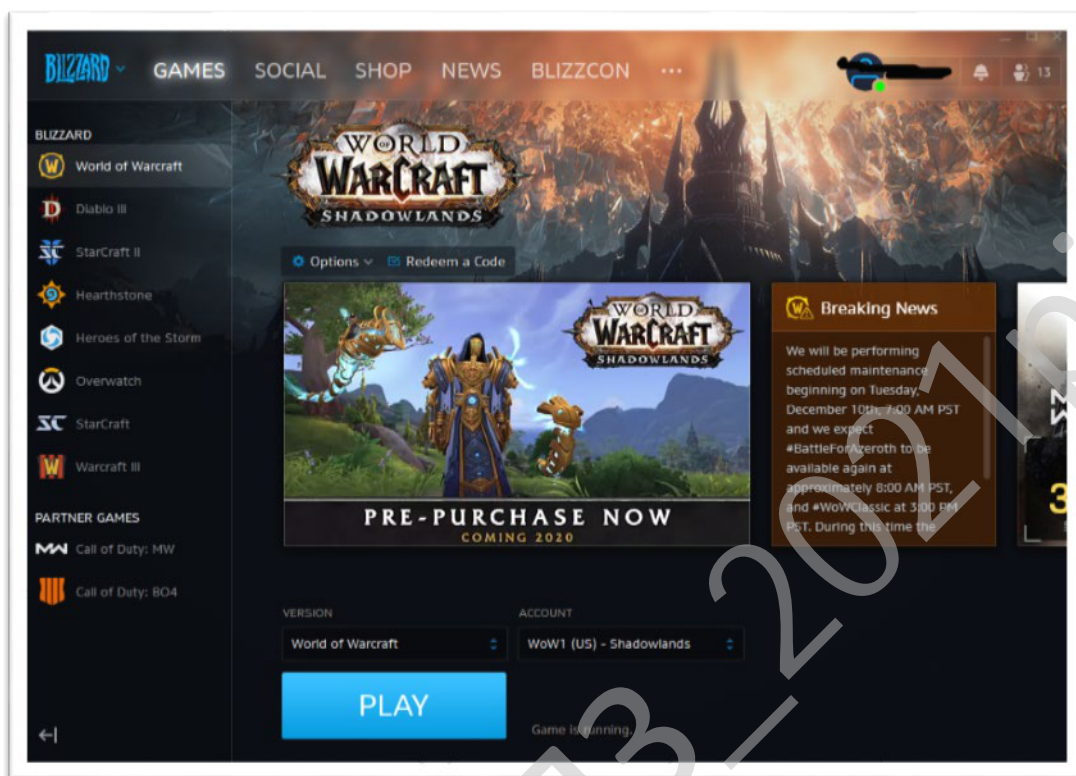


Рисунок 2.5 – Офіційний лаунчер World of Warcraft

На формі лаунчер можна побачити зліва в кутку розділ присвячений скачуванню різних ігор компанії Бліззард.

У верхній частині лаунчеру є переходи на різні вкладки:

- Ігри;
- Спільнота;
- Вбудований магазин;
- Новини.

У нижній частині лаунчеру можна вибрати версію гри та сервера на якому користувач буде грати

У центральній частині знаходяться слайди з різними новинами проекту

У лівій частині лаунчеру знаходиться кнопка повідомлення та кількість онлайн на серверах

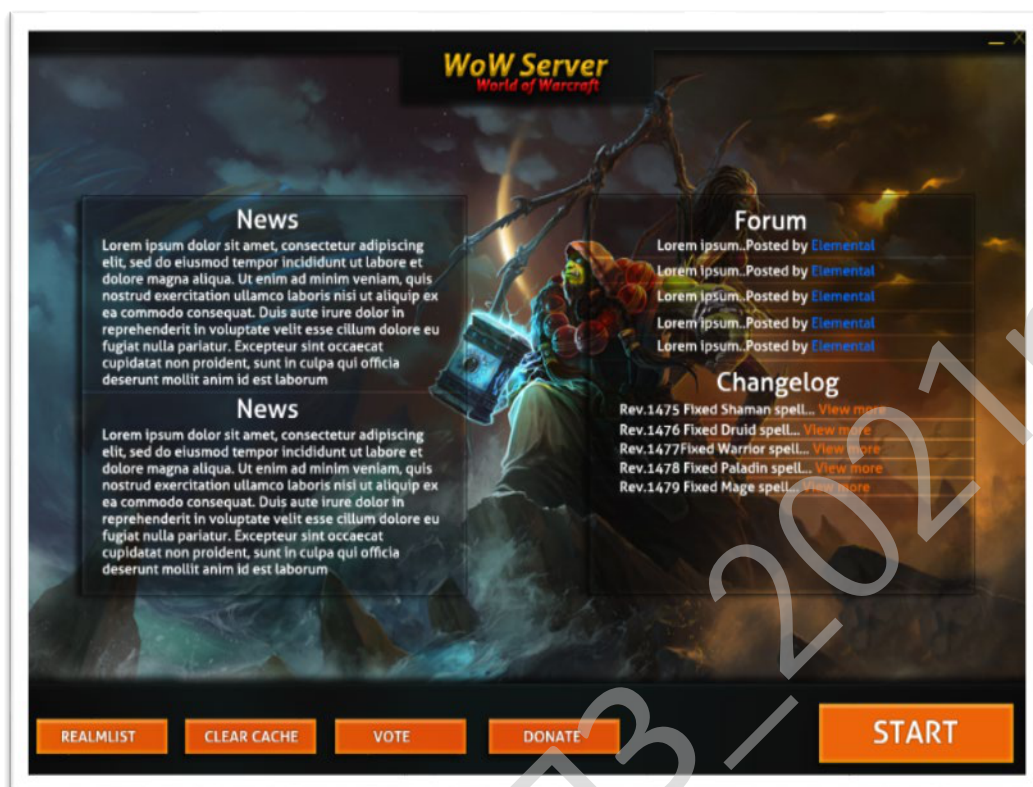


Рисунок 2.6 – Лаунчер проекту WoW-Server.org.

Є більш легкі лаунчери. Наприклад в даному додатку відсутня більшість функціоналу, такого як "Налаштування", "Перевірка клієнта", але є кнопка "Clear cache", яка дозволяє очистити кеш гри, також є кнопка "Realmlist", с допомогою якої можна відкрити список серверів і подивитися статистику і стан певного сервера.

Так само даний лаунчер відрізняється тим, що в ньому є можливість пожертвування грошей для підтримки проекту - "Donate" і функція оцінки сервера в загальній базі для визначення рейтингу проекту серед інших проектів гри.

Вим.	Арк.	№ докум.	Підпис	Дата

Розглянемо ще один популярний лаунчер "Arizona"

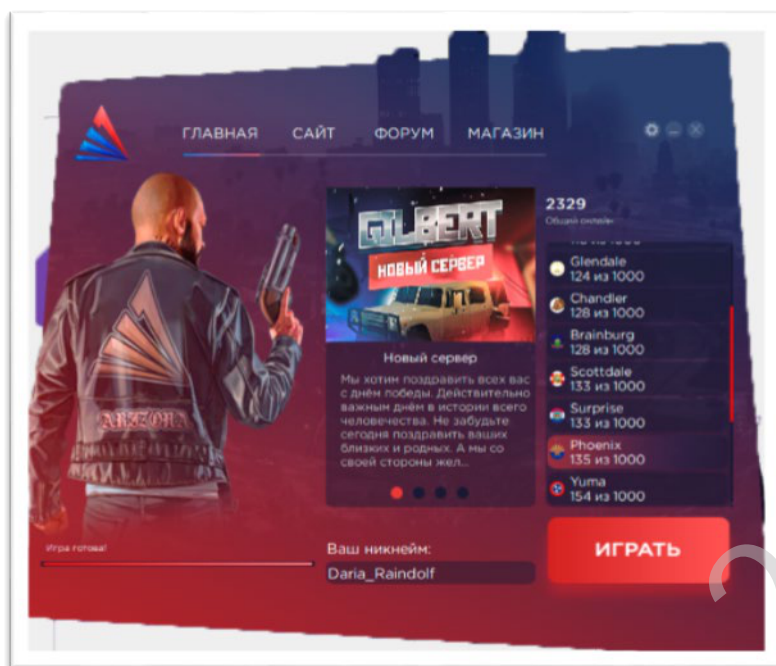


Рисунок 2.7 – Лаунчер проекту Arizona

Даний лаунчер досить інтуїтивно зрозумілий, у нього зверху є кілька вкладок:

Головна - відображає основне вікно лаунчера;

Сайт - направляє користувача на сайт проекту;

Форум - той же функціонал, як у кнопки "Сайт", тільки відправляє користувача на форум, де обговорюються ігрові та технічні аспекти гри / лаунчер.

Магазин - відкриває внутрішньоігровий магазин проекту, де люди можуть пожертвувати гроші, або купити спеціальні привілеї

У цього лаунчер так само присутнє вікно новин, з можливістю прокрутки серед 4х останніх публікацій.

Також є поле, для введення " Ніка " користувача. Ще присутній "прогрес-бар" який відображає статус завантаження оновлення / Фікса, він розташований внизу зліва. І кнопка грати, яка викликає гру.

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		16

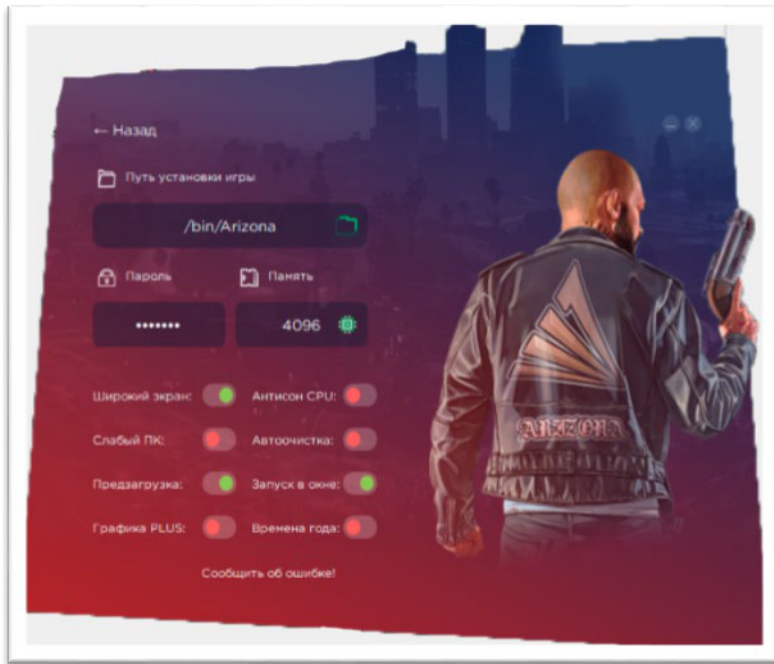


Рисунок 2.8 – Вкладка налаштувань Arizona

У самому верхньому правому куті розташована так само кнопка "налаштування" у вигляді шестерінки, вона відкриває вікно налаштувань, в даному вікні вже більш гнучкі налаштування під різні потужності комп'ютерів користувачів: від слабких до потужних.

Це вікно має настройку розташування гри, ще є вікно, в яке можна написати пароль від свого облікового запису і при заході у гру буде автоматично здійснюватися авторизація користувача. Так само можна збільшити або зменшити споживання оперативної пам'яті грою, вказавши в спеціальному вікні кількість доступних до споживання МБ.

Є перемикачі:

1. Широкий екран - дозволяє користувачів з широкоекранними моніторами більш комфортно відобразити інтерфейс в грі;
2. Слабкий ПК - дана можливість довантажує в грі текстури з низьким дозволом, вимикає ефекти, для більш високої продуктивності;
3. Перезавантаження – перезавантажує сутностей в пул;
4. Графіка PLUS - призначена для користувачів з більш потужними ПК, включає в себе текстури високого дозволу, розмиття,

покращені ефекти;

5. Антисон CPU - відключає тайм-аут між ітераціями в процесорі;

6. Авто-очищення - видаляє не передбачені розробниками модифікації гри;

7. Запуск у вікні - запускає гру У вікні, якщо з якихось причин гра не працює в повному екрані або для зручності стримінгу;

8. Пори року - додає в гру певні ефекти погоди, в залежності від поточного сезону.

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Для виконання дипломного проекту я використовував наступне програмне забезпечення:

Visual studio 2019 community – в першу чергу це безкоштовна версія середовища розробки Visual Studio. Щоб її використовувати, необхідно створити обліковий запис Visual Studio, в іншому випадку вона буде діяти 30 днів. Редакція має менший функціонал в порівнянні з платними редакціями, однак вона включає все необхідне для створення повноцінних додатків. Підходить для індивідуальних розробників і навчання, тому я вибрав саме цю платформу;

За допомогою Visual Studio можна розробляти:

– Класичні додатки для комп'ютера під керуванням операційної системи Windows;

– Мобільні додатки (Windows, iOS, Android);

– Web-додатки;

– Хмарні додатки;

– Різні розширення для Office, SharePoint, а також створення власних розширень для Visual Studio;

– Ігри;

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		18

- Бази даних SQL Server і SQL Azure.
- У Visual Studio Ви можете використовувати наступні технології і мови програмування: .NET, Node.js, C, C #, C ++, Python, Visual Basic, F #, JavaScript.
- Так як, в мене не дуже швидкий комп'ютер це теж вплинуло на те, що я використовував цю платформу для розробки додатка "лаунчер"
- Мінімальні вимоги до обладнання:
- Процесор з тактовою частотою не нижче 1,8 ГГц. Рекомендується використовувати як мінімум двоядерний процесор;
- ГБ оперативної пам'яті, рекомендується 4 ГБ (якщо встановлювати на віртуальну машину, то мінімум 2.5 ГБ);
- Вільного місця на жорсткому диску від 800 мегабайт до 100 гігабайт (Я використав приблизно 10-15 ГБ для того, щоб встановити потрібні бібліотеки), залежно від встановлених компонентів. У більшості випадків виділяйте як мінімум 30 гігабайт, наприклад, я встановлював всього кілька компонентів, і у мене зайняло більше 15 ГБ місця на жорсткому диску. Також Microsoft рекомендує встановлювати Visual Studio на SSD диск, я як раз встановлював на SSD (на своєму ноутбуці), швидкість роботи Visual Studio мене, звичайно, порадувала;
- Відеоадаптер з мінімальним дозволом 1280 на 720 пікселів (для оптимальної роботи Visual Studio рекомендується дозвіл 1366 на 768 пікселів і вище).

В новій версії Visual studio 2019 є багато нововведень. Нових можливостей в середовищі Visual Studio 2019 з'явилося дуже багато, я перерахую тільки невелику частину яку я використовував:

- Нові можливості установки оновлень - тепер можна вибрати спосіб установки оновлень Visual Studio і повністю контролювати його («Сервіс -> Параметри»), а також поновлення Visual Studio завантажуються у фоновому режимі, тільки коли комп'ютер не діє;

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		19

- Перероблений інтерфейс середовища, зокрема збільшено вертикальне простір для коду;
- Доданий вид розширення: «Безкоштовне», «Платне» і «Пробна версія» - тепер набагато легше розпізнати версію розширення, також додані відповідні теги, для більш зручного пошуку;
- Очищення коду одним клацанням - тепер відреагувати на попередження та рекомендації середовища можна одним натисканням кнопки;
- Вдосконалений пошук - тепер пошук видає більш точні результати (а також пошук став можливий в вікнах налагодження);

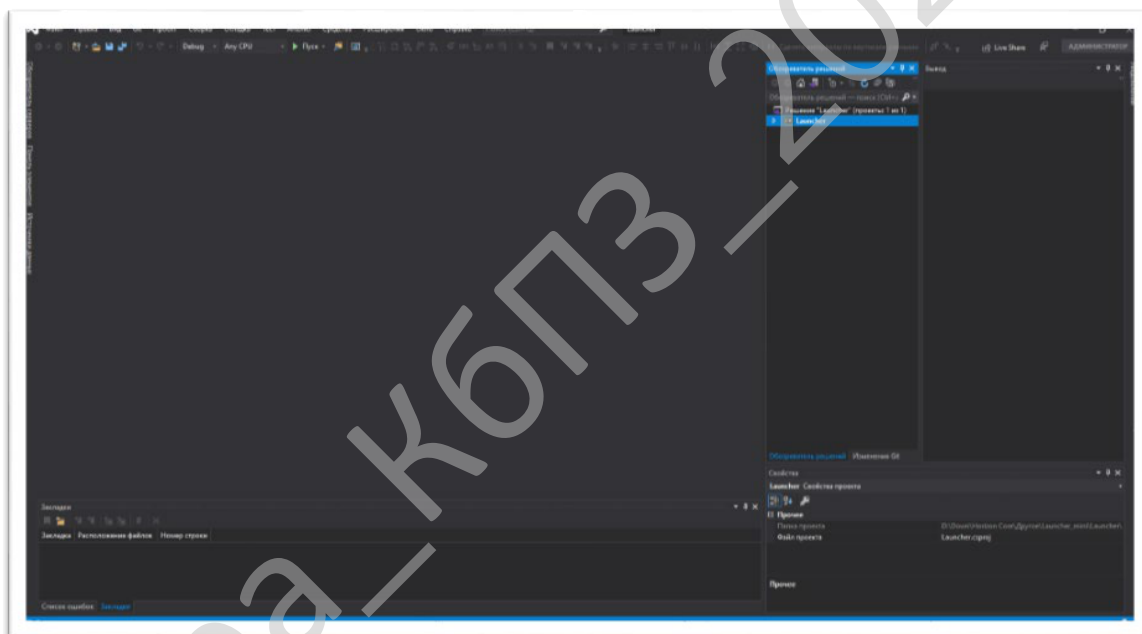


Рисунок 2.9 – Інтерфейс Visual Studio 2019

NotePad++ - це чудовий текстовий редактор з такими можливостями, яких не вистачало в Блокноті. До того ж, програмісти можуть його використовувати як редактор вихідного коду. Фактично ми можемо вважати його одним з найкращих десяти безкоштовних додатків, які повинні бути встановлені на комп'ютері.

Серед його функцій можна знайти:

- Кольорове підсвічування синтаксису для різних мов програмування.

- WYSIWYG (What You See Is What You Get), що дозволяє вам бачити в точності те, що буде надруковано.
- Можна вибрати колір для кожного з підтримуваних мов програмування, завдяки чому ви зможете їх розрізнити з першого погляду.
- Можна переглядати одночасно кілька документів.
- Програма підтримує C, C++, Java, XML, HTML, PHP, JavaScript, makefile, файли з ASCII графікою, doxygen, ini, ASP, VB / VBS, SQL, Objective-C, CSS, Pascal, Perl і Python.

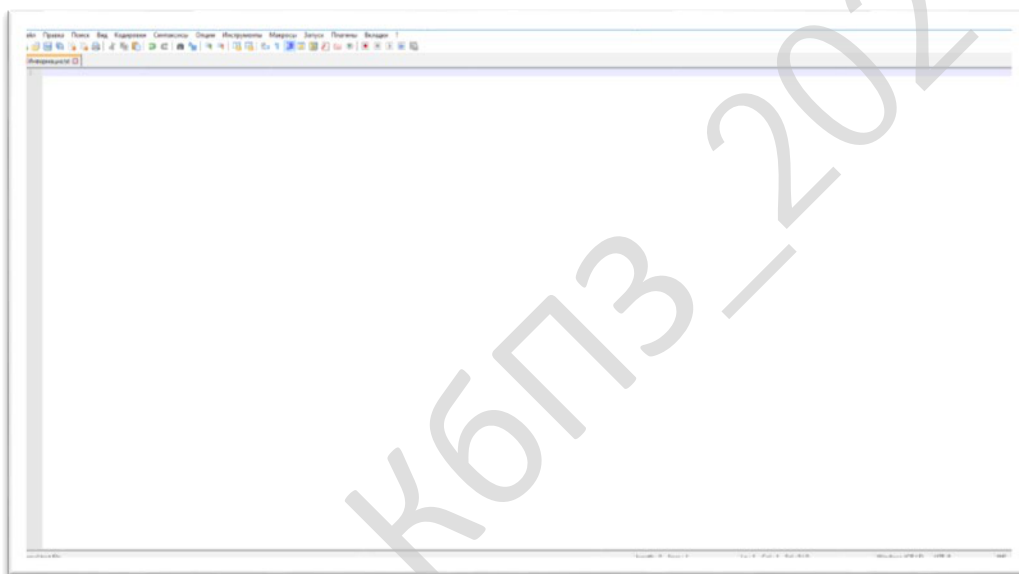


Рисунок 2.10 – Інтерфейс Notepad++

Мова програмування на якій я писав свій додаток – це C#.

C# - сучасна об'єктно-орієнтована і типобезпечна мова програмування. C# дозволяє розробникам створювати безліч типів безпечних і надійних програм, які потребують екосистемі .NET. C# відноситься до широко відомому сімейству мов C, і здасться добре знайомим кожному, хто працював з C, C++, Java або JavaScript.

Програми C# виконуються в .NET, віртуальній системі виконання, що викликає загальномовного середовища виконання (CLR) і набір бібліотек класів. Середовище CLR - це реалізація загальномовної інфраструктури мови (CLI), що є

міжнародним стандартом, від корпорації Майкрософт. CLI є основою для створення середовищ виконання і розробки, в яких мови і бібліотеки прозоро працюють один з одним.

У C# існують два різні види типів: довідкові типи і типи значень. Змінні типу значень містять безпосередньо дані, а в змінних посилальних типів зберігаються посилання на потрібні дані, які іменуються об'єктами. Дві змінні посилального типу можуть посилатися на один і той же об'єкт, тому може статися так, що операції над однією змінною торкнуться об'єкт, на який посилається інша змінна. Кожна змінна типу значення має власну копію даних, і операції над однією змінної не можуть зачіпати іншу (за винятком змінних параметрів `ref` і `out`).

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		22

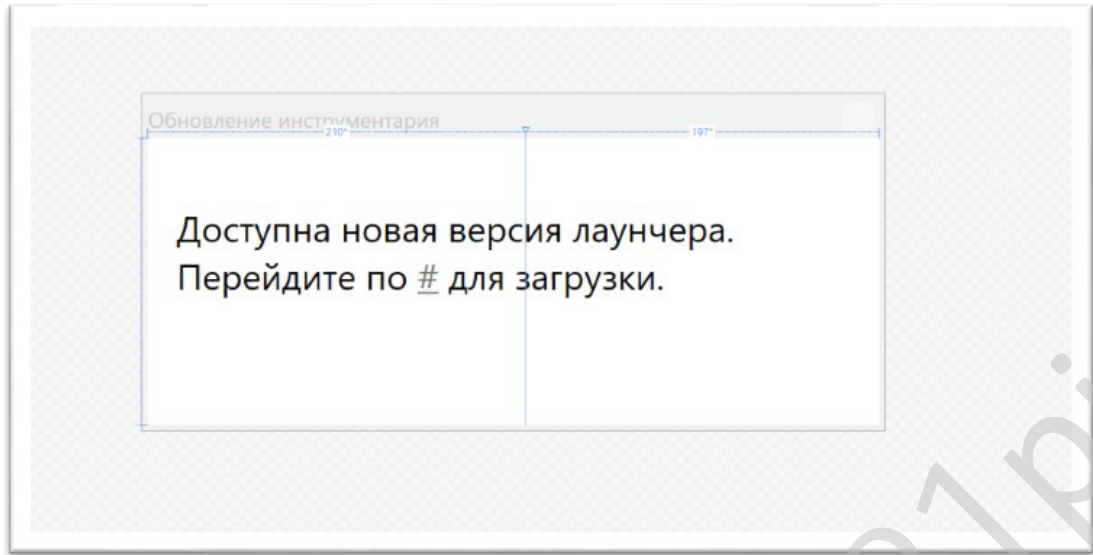


Рисунок 3.1 - Конструктор формы LNewVer.xaml

Другий файл MainWindow.cs – створює головну форму додатку та інтерфейс функціонування.

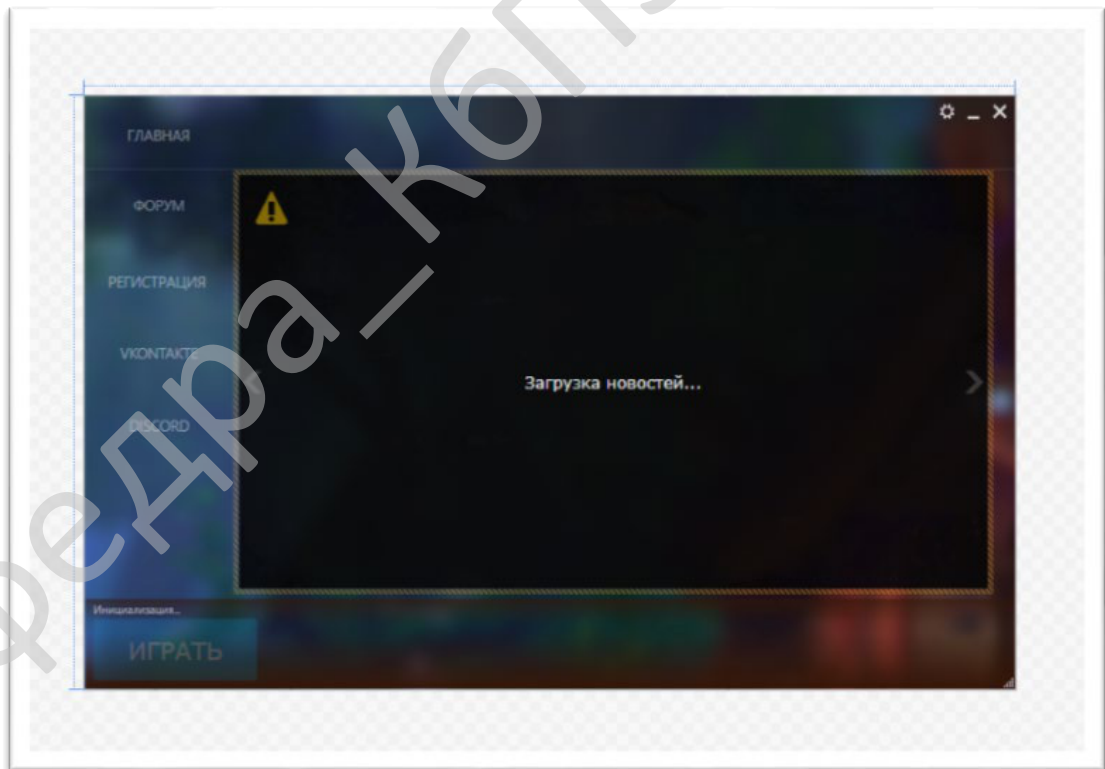


Рисунок 3.2 - Конструктор формы MainWindow.cs

Вим.	Арк.	№ докум.	Підпис	Дата

КБР-123.21.0051.00.00.ПЗ

Арк.

24


```

{
    Description = @"Выберите папку с клиентом игры",
    RootFolder = Environment.SpecialFolder.MyComputer,
    ShowNewFolderButton = false
};
var result = folder.ShowDialog();
if (result != System.Windows.Forms.DialogResult.OK) return;

if (File.Exists(Path.Combine(folder.SelectedPath, "Wow.exe")))
{
    var folderPath = folder.SelectedPath;
    Folder.Text = folderPath;
}
else
{
    MessageBox.Show("Файл \"Wow.exe\" не найден!\nПожалуйста
выберите корректную папку с игрой!", "Ошибка выбора папки", MessageBoxButtons.OK,
MessageBoxImage.Error);
}
}

```

В даному меню настройки клієнта гри, потрібно вказати шлях до папки з грою, щоб лаунчер відкривався без помилок.

Друга функція – автоматична авторизація, при відкритті клієнта гри автоматично заповнює поля (логін та пароль) - це забезпечую невелику вільну частину часу.

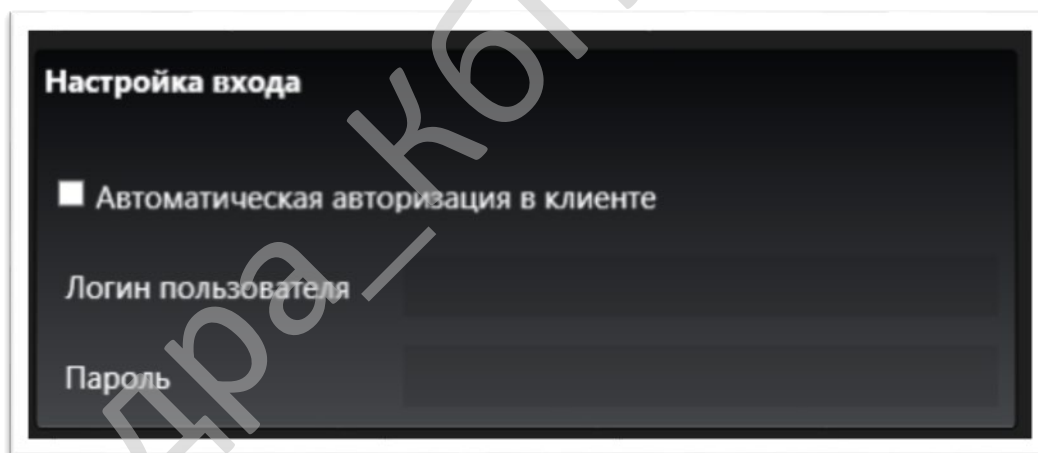


Рисунок 3.6 - Меню автоматичної авторизації

Користувач вводить логін та пароль, після цього при натисканні кнопки «грати» поля в клієнті гри будуть заповнюватися автоматично (не потребується ручний ввід логіна та паролю).

Дана функція абсолютно безпечна, так як логін і пароль нікуди не записується та не відправляється на хостинг сервера.

Третя функція – видалення серверних файлів с клієнта. Необхідна для того, якщо гравець більше не хоче грати на сервері та не хоче встановлювати знову новий клієн гри, лаунчер автоматично видалить всі файли та клієнт гри прийме стандартний вигляд.

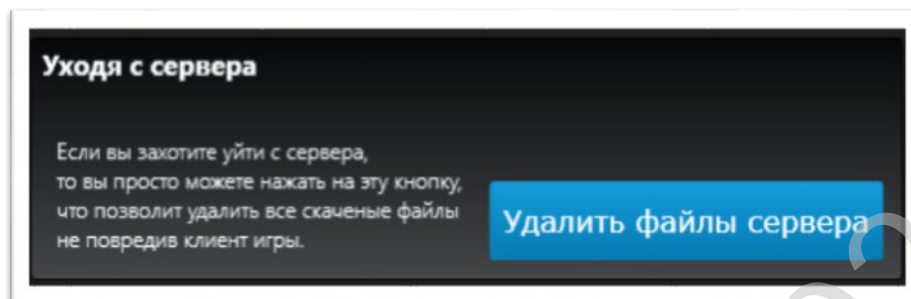


Рисунок 3.7 - Меню видалення файлів сервера

```

/// <summary>
/// Код видалення файлів серверу
/// </summary>
private void ResetPath_Click(object sender, RoutedEventArgs e)
{
    Folder.Text = "Не задано";
}

private void ButtonDelete_Click(object sender, RoutedEventArgs e)
{
    if (MessageBox.Show("Вы действительно хотите удалить все загруженные файлы сервера?", "Подтверждение удаления", MessageBoxButton.YesNoCancel, MessageBoxImage.Question) == MessageBoxResult.Yes){
        if (Owner is MainWindow m) m.DPatches();
    }
}

/// <summary>
/// Код виведення інформації про видалення файлів
/// </summary>
File.Delete(_updatePath);
MessageBox.Show("Все файлы успешно удалены", "Удаление файлов", MessageBoxButton.OK, MessageBoxImage.Information);
btn_play.Visibility = Visibility.Hidden;
labelmsg.Content = "Запуск клиента невозможен";

```

Останнє меню – налаштування загрузки. Можна вказати:

1. Варіант відображення загрузки
2. Варіанти відображення значення загрузки (кб/с, мб/с, гб/с)

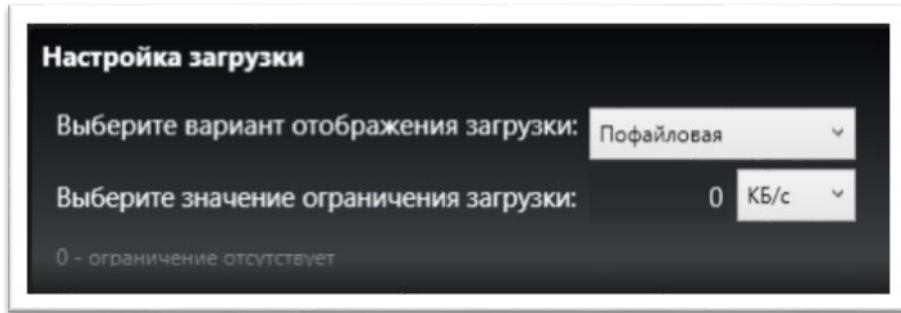


Рисунок 3.8 - Меню налаштування загрузки

Наступне меню слайдів – потрібно для виведення основних новин проекту

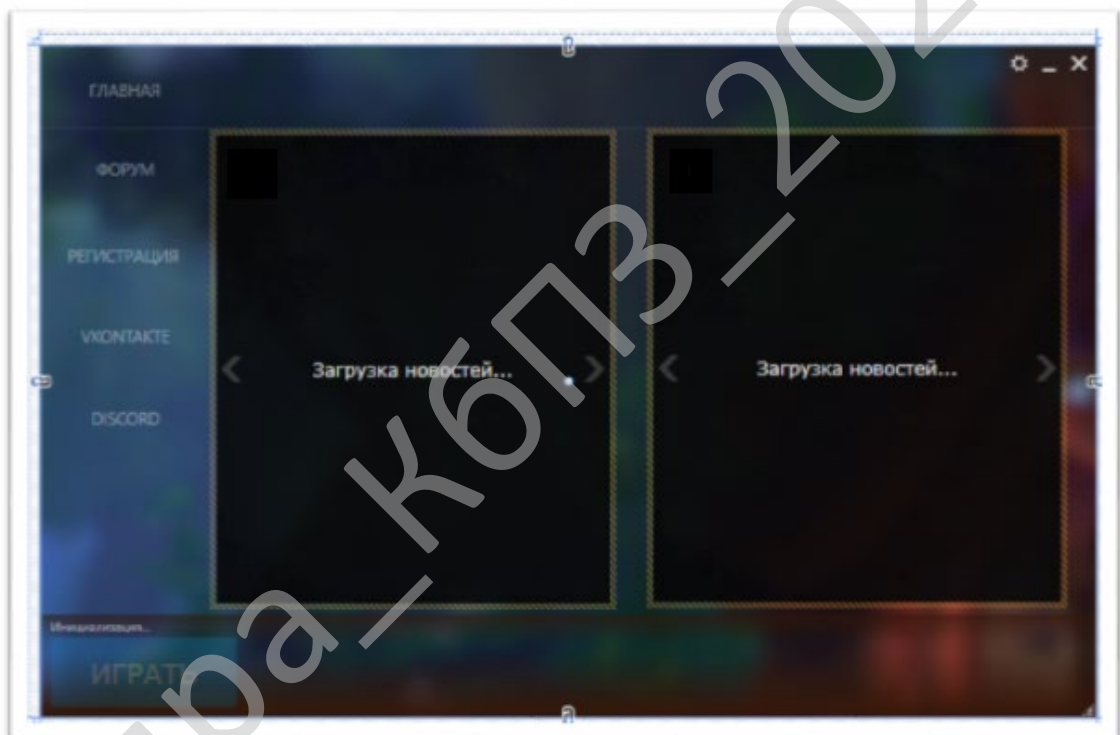


Рисунок 3.9 - Меню слайдів

```

/// <summary>
/// Код зміни відображення новин
/// </summary>
private void ChangeNewsItem(bool next)
{
    if (next)
    {
        _node = _node.Next ?? _newsList.First;
    }
    else
    {
        _node = _node.Previous ?? _newsList.Last;
    }
}

```

Вим.	Арк.	№ докум.	Підпис	Дата

КБР-123.21.0051.00.00.ПЗ

Арк.

31

```

        var sb = FindResource("ChangeItemsBegin") as Storyboard;
        Storyboard.SetTarget(sb ?? throw new
InvalidOperationException("Не знайдено ресурсу \"ChangeItemsBegin\""), MainGrid);
        sb.Completed += StoryBoard_Completed;
        sb.Begin();
    }

    /// <summary>
    /// Код XML меню слайдів
    /// </summary>
    <local:NewsLoaderControl x:Name="news_box" Margin="170,10,30,10"
Grid.Row="1" Foreground="White" Loaded="news_box_Loaded" Grid.ColumnSpan="2"/>
        <Label x:Name="contentNewsLoadIndicator" Grid.Row="1"
Margin="1,0,1,0" Foreground="White" HorizontalContentAlignment="Center"
VerticalContentAlignment="Center" Visibility="Hidden" Grid.ColumnSpan="2">
            <TextBlock x:Name="newsLoadIndicator"
TextWrapping="Wrap" FontSize="12pt"><Run Text="Завантаження новостей..."></TextBlock>
        </Label>

```

І саме головне переходить на веб-сторінки проекту:

- Головна – перехід на основну сторінку;
- Форум – перехід на сторінку форуму;
- Реєстрація – перехід на сторінку реєстрації;
- Facebook – перехід на соціальну сторінку;
- Discord – перехід на канал проекту.

```

    /// <summary>
    /// Код посилання на сторінки
    /// </summary>
    private void link_main_Click(object sender, RoutedEventArgs e)
    {
        Process.Start("http://horizon-world.net");
    }
    private void link_cabinet_Click(object sender, RoutedEventArgs e)
    {
        Process.Start("http://horizon-world.net/forum");
    }
    private void link_registration_Click(object sender, RoutedEventArgs e)
    {
        Process.Start("http://horizon-world.net/reg");
    }
    private void link_social_Click(object sender, RoutedEventArgs e)
    {
        Process.Start("https://facebook.com/horizon_wow");
    }
    private void link_social_Copy_Click(object sender, RoutedEventArgs e)
    {
        Process.Start("http://horizon-world.net/minecraft");
    }

    private void link_social_Copy1_Click(object sender, RoutedEventArgs e)
    {
        Process.Start("https://discord.com/invite/hkMzUPaMuJ");
    }

```

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		32

3.2 Розробка структурної схеми

Структурна схема розробленої системи зображена на рисунку 3.10. На ній показано структуру.

У складанні структури додатку лаунчера важливо враховувати доступність та зручність навігації для користувача. Кожний розділ розбитий на більш-менш рівну кількість розділів.

Головне меню включає основні розділи. Якщо максимально спростити структуру – це збільшить кількість позитивних відгуків від користувачів. Але якщо в додатку буде багато зайвої інформації - користувачі відразу відчують незручність і навіть роздратування.

На структурній схемі додатку ми можемо бачити які є функції і що нам відкривається для доступного використання:

Перші можливості – перехід на веб-сторінки:

– Головна – відкриває веб-сторінку проекту на якому відображена вся інформація проекту;

– Форум – відкриває веб-сторінку спільноти проекту де гравці можуть спілкуватися ;

– Реєстрація – відкриває сторінку реєстрації для гри на проекті, та для автоматичної авторизації через додаток (в майбутньому планується додати в лаунчер вбудовану функцію реєстрації);

– Facebook – відкриває соціальну.

– Discord – відкриває додаток «Discord» для вступу в канал проекту (Якщо на Windows не встановлена ця програма, тоді відкриється браузер).

Наступна функція «Запуск клієнта гри» - Кнопка відповідає за вмикання клієнта гри (Якщо лаунчер оновлює клієнт і почалася загрузка патчів кнопка буде заблокована и матиме сірий кольор, поки не закінчиться скачування оновлення).

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		33

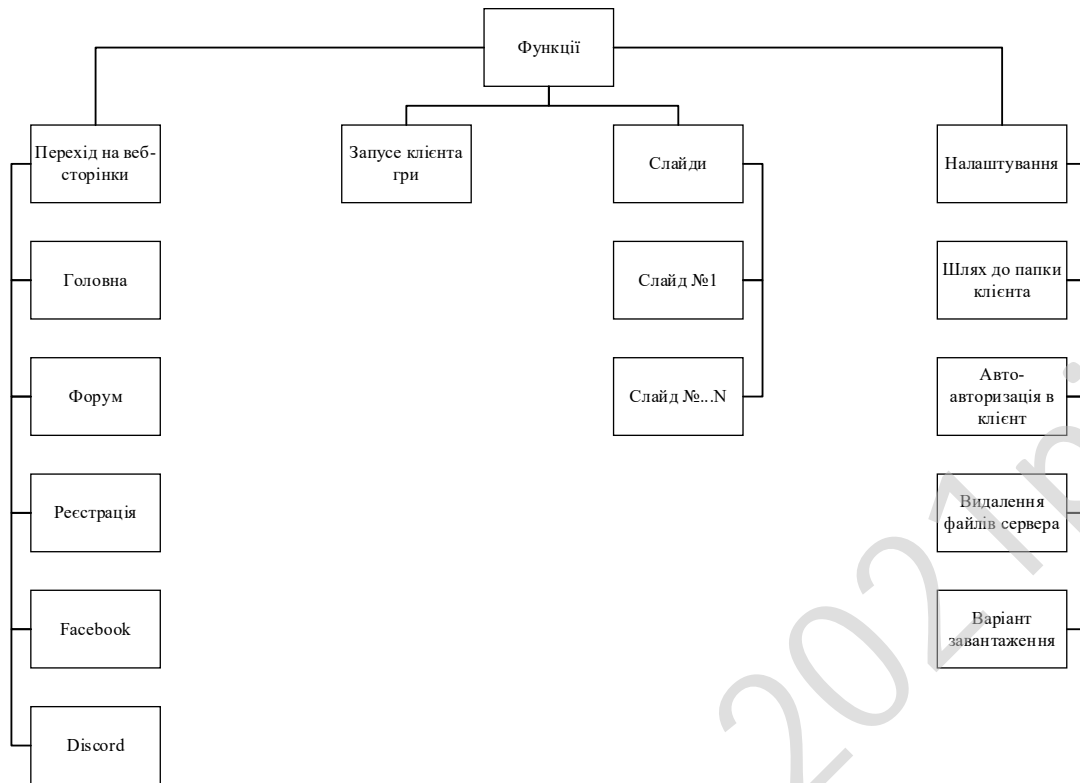


Рисунок 3.10 – Структурна схема додатку

3.3 Розробка функціональної схеми

Функціональна схема розробленої системи зображена на рисунку 3.11.

З рисунку видно, що розроблена система складається з наступних частин:

Функціональна схема - умовне графічне зображення, яке відображає функції, виконувані окремими елементами системи й зв'язки між ними. Елементи функціональної схеми зображують у вигляді прямокутників.

Функціональна схема в порівнянні з структурною більш детально розкриває функції окремих елементів і пристроїв.

Функціональна схема є основним технічним документом, що визначає функціонально-блокову структуру окремих вузлів, управління і регулювання технологічного процесу і оснащення об'єкта управління приладами і засобами автоматизації.

Запуск клієнта гри – при натисканні кнопки відкривається клієнт гри, та після його закриття повертається до додатку.

Наступна дія слайди – як показано на діаграми слайди можна продивлятися та повертатися до початку додатку.

Вікно налаштувань – відкриває вікно налаштування в якому користувач може встановити відповідні налаштування та повернутися до головної форми додатку не перезавантаживши його.

Таким чином, розглянувши опис системи, структурну, функціональну схеми системи, та діаграму взаємодії процесів перейдемо до опису блок-схем основної програми, та підпрограм, які використовуються, для реалізації системи.

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		37

4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ

4.1 Блок-схеми та опис алгоритмів функціонування системи

Розглянемо алгоритм роботи основної програми. Його блок-схема зображена на рисунку 4.1.

З рисунку видно, що після запуску програми спочатку відбувається вивід основного вікна програми.

Алгоритм і блок-схема є два типи інструментів для пояснення процесу програми. Алгоритми і блок-схеми - це два різних інструменту, використовуваних для створення нових програм, в комп'ютерному програмуванні. Алгоритм представляє собою покроковий аналіз процесу, в той час як блок-схема пояснює кроки програми в графічному вигляді.

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		38

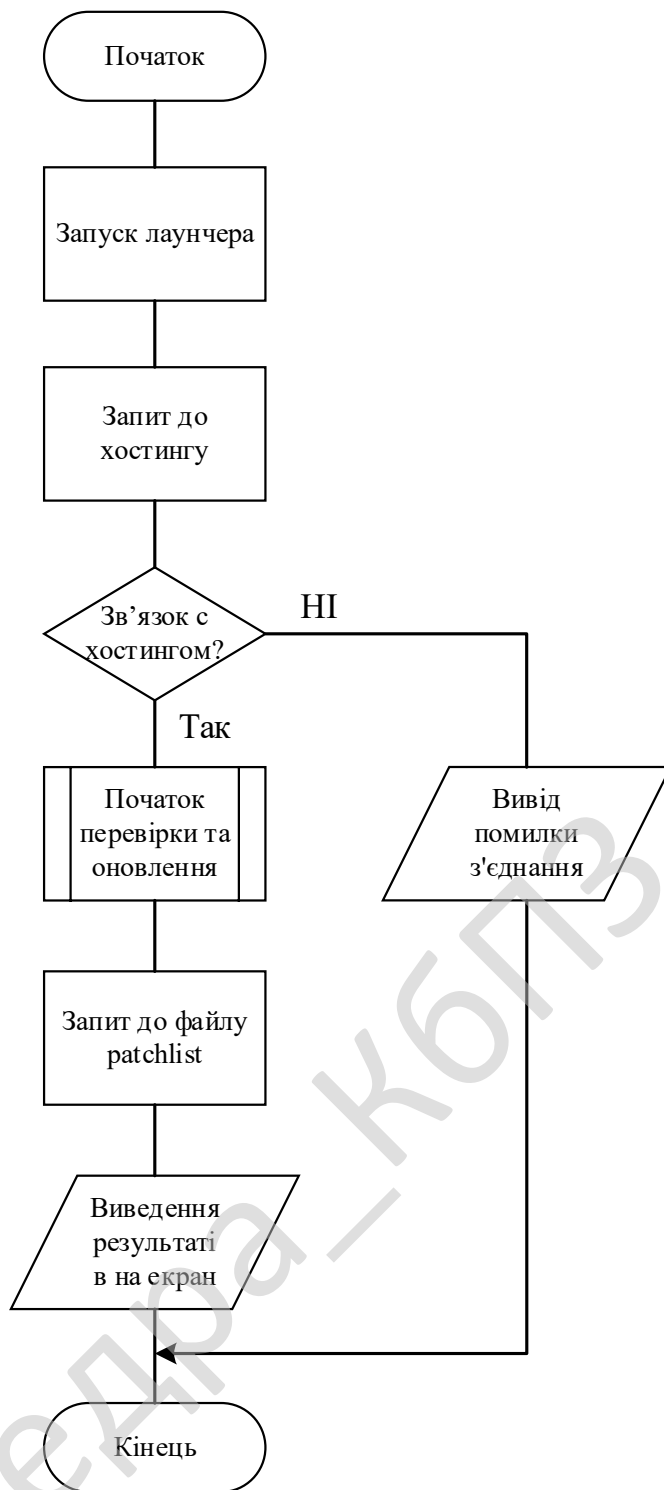


Рисунок 4.1 - Блок-схема алгоритму запуску лаунчеру

Видно що коли запускається лаунчер та йде запит до хостингу-сайту на якому знаходиться файл patchlist, в якому зберігається інформація про файли які необхідно перевірити та оновити.

Наступна дія перехід на сторінку реєстрації:

```
/// <summary>
/// Код XML переходу на сторінку «Реєстрація»
/// </summary>
<Button x:Name="link_cabinet" Content="ФОРУМ" HorizontalAlignment="Left"
Margin="24,3,0,391" Style="{DynamicResource LinkButton}" Width="115" FontSize="16"
Click="link_cabinet_Click" Grid.Row="1"/>

/// <summary>
/// Код посилання на кнопку «Головна»
/// </summary>
private void link_registration_Click(object sender, RoutedEventArgs e)
{
}

/// <summary>
/// Код початок процесу при натисканні кнопки link_registration_Click
/// </summary>
private void link_registration_Click(object sender, RoutedEventArgs e)
{
    Process.Start("http://horizon-world.net/reg");
}
}
```

Наступні дії аналогічного переходу аж до дії "Налаштування", в виборі налаштування відкривається вікно підпрограми «налаштування», можна подивитися на рис. 4.3. Там наглядно показано як робить підпрограма.

Наступна дія запуск клієнта гри при натисканні на кнопку «грати» запускається клієнт котрий був вказаний у налаштуванні шляху клієнта.

```
/// <summary>
/// Код XML кнопки «грати»
/// </summary>
<Button x:Name="btn_play" Grid.Row="2" Margin="10,0,0,10"
Click="btn_play_Click" Height="67" VerticalAlignment="Bottom"
HorizontalAlignment="Left" Width="180" IsTabStop="False" Content="ИГРАТЬ"
Style="{DynamicResource PlayButton}" FontFamily="Verdana" FontSize="26.667"
Cursor="{Binding Cursor, ElementName=version}" BorderThickness="0"/>

/// <summary>
/// Код початок процесу при натисканні кнопки TaskbarPlay_Click
/// </summary>
private void TaskbarPlay_Click(object sender, EventArgs e)
{
    Play();
}
}
```

Та остання дія – закриття лаунчеру, при натисканні якої вимикається лаунчер.

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		41

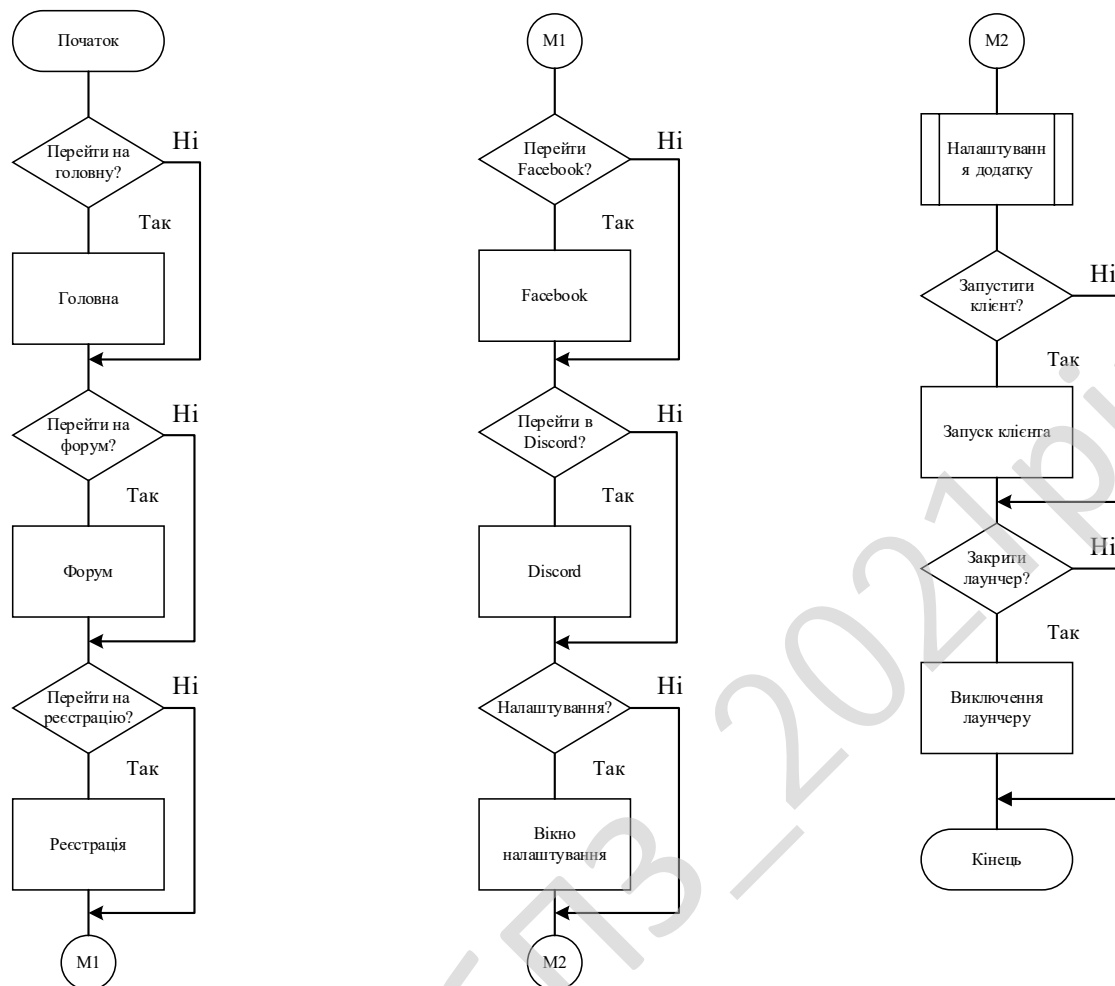


Рисунок 4.2 - Блок-схема алгоритму лаунчеру

Підпрограма на рис. 4.3. показує як робить вікно налаштування лаунчеру і які функції вона виконує

Як видно з блок-схеми є чотири умови:

Перша – «Налаштування клієнта?», якщо ми вибираємо цей пункт тоді задаємо шлях до папки де знаходиться гра, після чого при запусканні лаунчеру він буде знати з якої папки треба запускати гру при натисканні кнопки «грати». Після того як ми виконали даний пункт можемо перейти до наступного, - якщо не вибрали налаштування клієнта тоді переходимо одразу до «Авторизації клієнту».

```

/// <summary>
/// Код налаштування шляху до клієнту гри
/// </summary>
private void Button_Click_1(object sender, RoutedEventArgs e)
{
    var folder = new System.Windows.Forms.FolderBrowserDialog
    {
        Description = @"Выберите папку с клиентом игры",
    }
}

```

Вим.	Арк.	№ докум.	Підпис	Дата
------	------	----------	--------	------

КБР-123.21.0051.00.00.ПЗ

Арк.

42

}
}

Та останній пункт «налаштування загрузки» - вводиться варіант загрузки (швидкість загрузки, поетапний вид загрузки, змішаний або спільний, який буде показуватися при оновленні лаунчером.

Кафедра _ КБПЗ _ 2021 рік

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		44

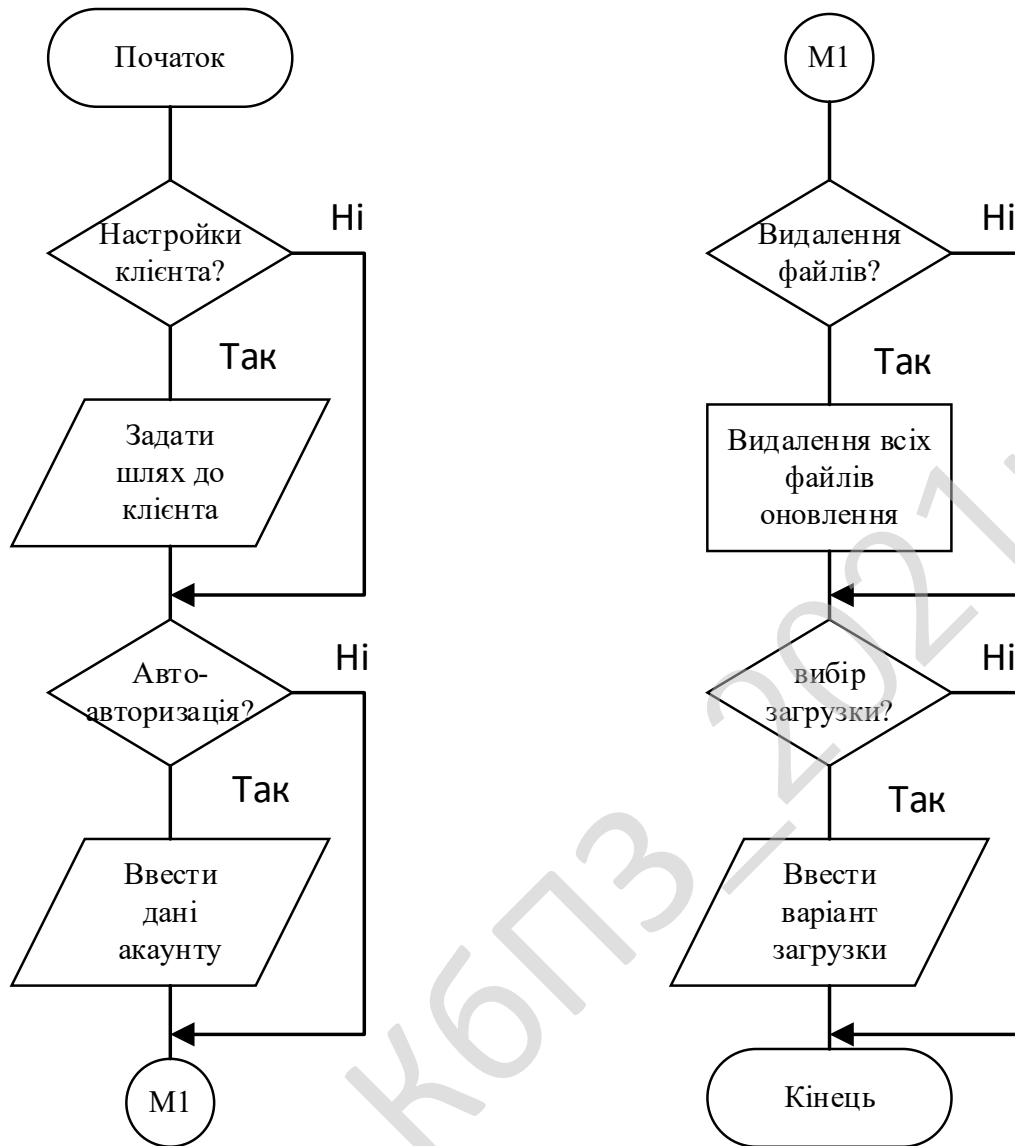


Рисунок 4.3 - Блок-схеми роботи підпрограми налаштування

4.2 Захист розробленого програмного забезпечення

Головна проблема .NET програм, так само як Java - легкий спосіб отримання початкових кодів.

Існує дуже багато протекторів, таких як: Themida, Confuser, Enigma, VMProtect, .NET Reactor

Я буду використовувати ПО .NET Reactor (рис. 4.4)

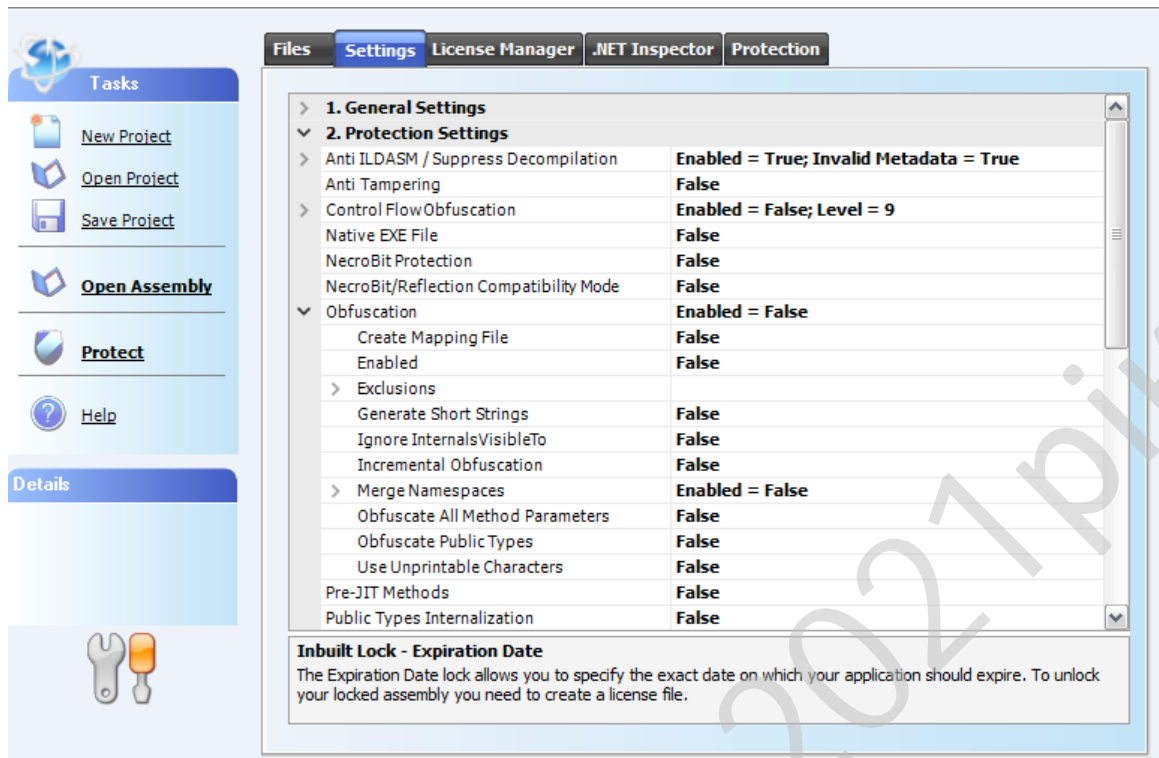


Рисунок 4.5 – Інтерфейс налаштувань .NET Reactor

Коли все налаштували, тискаємо на кнопку Protect! Починає йти процес захисту додатку.

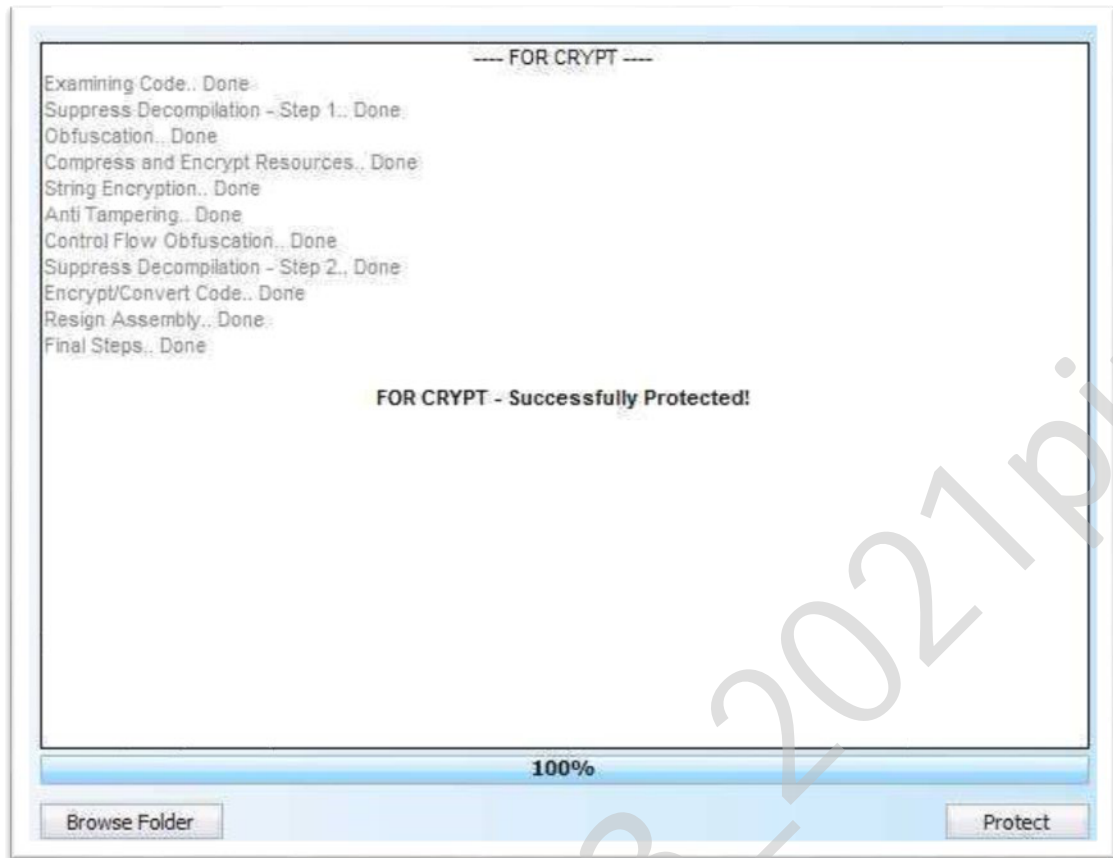


Рисунок 4.6 – Процес захисту додатку

Додаток повністю працює, і розшифрувати цей захист зможе тільки професіонал, через який-небудь debugger.

Питання захисту інтелектуальної власності стояв завжди перед розробниками. Піратство і Антипіратство буквально крокують рука об руку, створюючи всілякі інструменти. С # додатки були спочатку вразливі для реверсінга. У цьому плані, вони не сильно далеко пішли навіть від того ж JS, хоча там взагалі все очевидно.

Тим не менш, треба не забувати, що будь -яку програму завжди можна зламати. Питання стоїть тільки в тому, кому це знадобиться.

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

5.1 Експлуатація і індивідуальний підхід до промислової експлуатації

На даному етапі виконується проект впровадження системи ігрового лаунчера для гри World of Warcraft в промислову експлуатацію. А саме на ігровий проект Horizon-World, де програма буде виконувати всі свої функції.

На етапі введення системи в промислову експлуатацію виконується перехід до використання таких завдань як налаштування файлів на хостинг серверу.

Треба загрузити наступні файли:

News – відповідає за виведення новин в додатку Лаунчер.

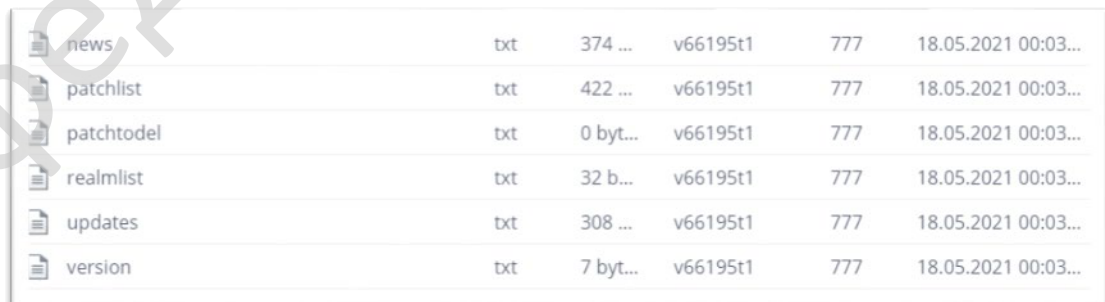
Patchlist – в даному файлі треба вводити всі назви патчів для оновлення клієнту гри та обов'язково посилання на файли потрібні містити прямі посилання. Наприклад:

<http://horizon-world.net/patches/Wow.exe#Wow.exe>

<http://horizon-world.net/patches/blizzlike/patch-ruRU-m.mpq>

<http://horizon-world.net/patches/blizzlike/patch-ruRU-7.mpq>

<http://horizon-world.net/patches/blizzlike/patch-G.mpq>



news	txt	374 ...	v66195t1	777	18.05.2021 00:03...
patchlist	txt	422 ...	v66195t1	777	18.05.2021 00:03...
patchtodel	txt	0 byt...	v66195t1	777	18.05.2021 00:03...
realmist	txt	32 b...	v66195t1	777	18.05.2021 00:03...
updates	txt	308 ...	v66195t1	777	18.05.2021 00:03...
version	txt	7 byt...	v66195t1	777	18.05.2021 00:03...

Рисунок 5.1 - Файли для роботи з лаунчером

Patchodel – Файл відповідає за видалення патчів с клієнту які раніше були встановлені на ігровий сервер.

Updates – файл містить інформацію про оновлення лаунчеру.

Version – файл відповідає за версію лаунчеру, якщо файл змінити тоді буде видавати вікно з тим, що вийшла нова версія лаунчера з посиланням.

5.2 Інструкція користувача

На рис. 5.2 показаний лаунчер дипломного проекту, який в знаходиться в кінцевій стадії розробки.

На ньому показані функції програми, такі як:

1. Кнопки переходу на веб-ресурси;
2. Кнопка запуску клієнта гри;
3. Інформація під час оновлення або скачування патчів (Показує % скачування, швидкість скачування оновлення файлів гри);
4. Кнопка яка відкриває вікно налаштувань.



Рисунок 5.2 - Лаунчер дипломного проекта

Наступний рис. 5.3 - на якому показане вікно налаштувань, в якому можна виконати такі перелічені функції:

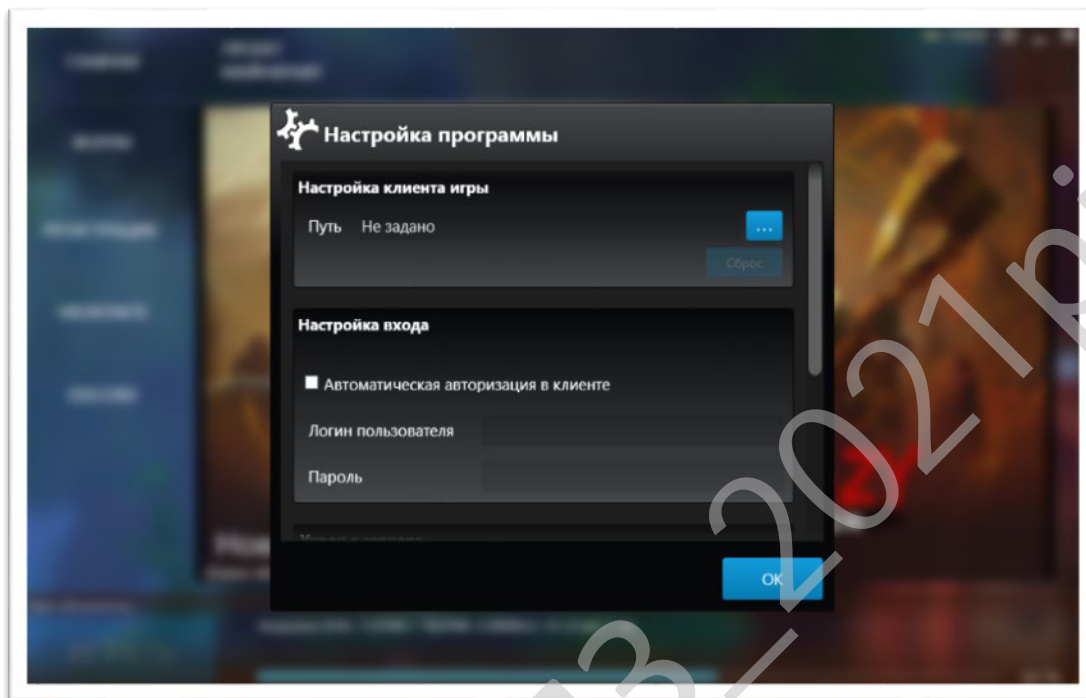


Рисунок 5.3 - Вікно налаштувань

5.3 Інструкція інженера

У цьому посібнику мною будуть розглянуті основні моменти по налаштуванню і подальшої експлуатації «лаунчер». Основна настройка не відніме багато часу.

Основна настройка вихідних кодів програми перед компіляцією

Properties

Перше з чого ми почнемо - це розділ Properties на панелі «Оглядач рішень». На рис.5.4 показано де знаходиться даний розділ.

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		51

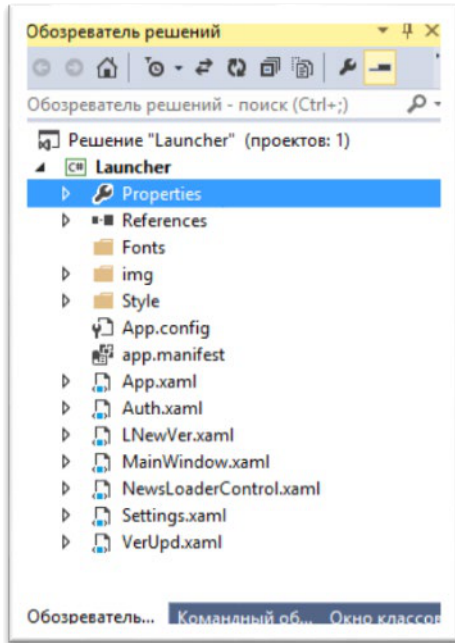


Рисунок 5.4 - «Properties» на панелі «Оглядач рішень»

Клацніть по ньому двічі, щоб відкрити вікно основних налаштувань рішення. На рис.5.5 продемонстрована відкрита вкладка, де і здійснюються основні настройки параметрів і властивостей збірки або рішення в цілому.

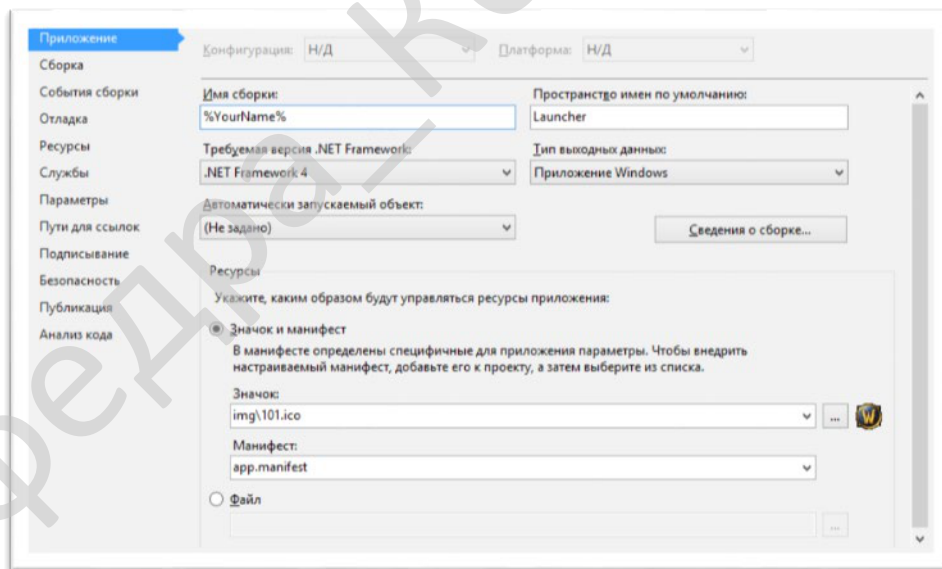


Рисунок 5.5 - Початковий екран налаштувань

Змінити значення поля «Ім'я збірки» на назву вашого лаунчер. Клацнути по кнопці «Відомості про збірку», щоб відкрити додаткові параметри властивостей майбутнього програми.

Сведения о сборке

Название: Horizon

Описание: Программа запуска World of Warcraft

Организация: Horizon Lab

Продукт: Horizon Warcraft Launcher

Авторские права: Copyright © Casper 2021

Товарный знак: Casper World of Warcraft Launchers

Версия сборки: 1 0 0 0

Версия файла: 1 0 0 0

Идентификатор GUID: F37E84CB-D76A-49B1-A1AC-88879903087B

Нейтральный язык: (Нет)

Сделать сборку видимой для COM

OK Отмена

Рисунок 5.7 - Додаткові настройки відомостей про збірку

Змінити значення полів «Назва» та «Продукт» згідно імені лаунчера або сервера. Натиснути «ОК» для підтвердження змін. На цьому робота у вкладці «Додаток» закінчена».

Перейти у вкладку «Параметри».

Вим.	Арк.	№ докум.	Підпис	Дата

КБР-123.21.0051.00.00.ПЗ

Арк.

53

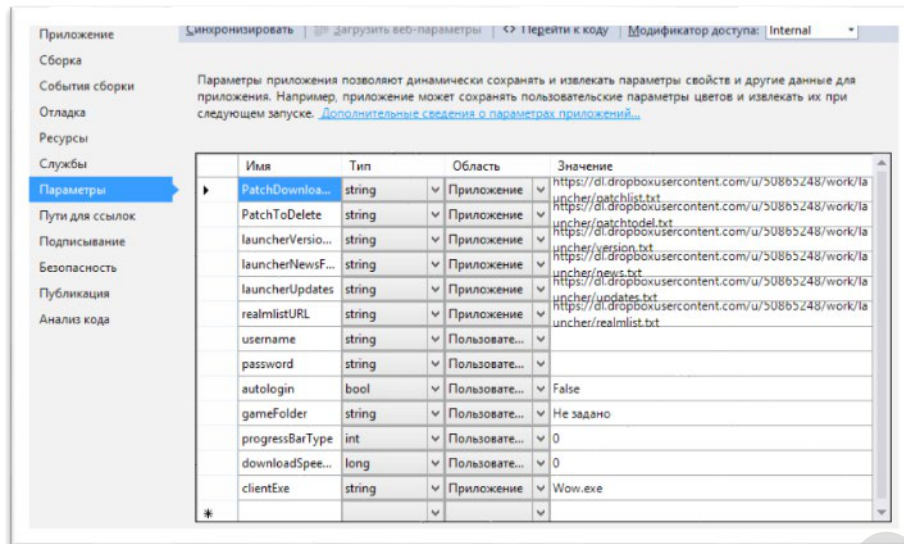


Рисунок 5.8 - Зовнішній вигляд вкладки «Параметри»

Це глобальні змінні конфігурації програми. Зміни значень підлягають тільки ті змінні, область яких поширюється на додаток.

Докладніше:

- PatchDownloadURL - Список посилань оновлень для клієнта;
- PatchToDelete - Посилання на список імен файлів, що підлягають видаленню (Сезонні (новий рік) або невдало поновлення, які встигли завантажити). Регістр букв має значення;
- LauncherVersionUrl - Посилання на файл з актуальною версією програми;
- LauncherNewsFileUrl - Посилання на файл новин для програми;
- LauncherUpdates - Посилання на файл з історією оновлень і змін в програмі;
- RealmistURL - Посилання на файл зберігає інформацію про реалмліст для підключення до сервера;
- ClientExe - Ім'я запускаемого об'єкта (стандартний або пропатчений .exe файл гри).

Єдина вимога до посилань - вони повинні бути прямими.

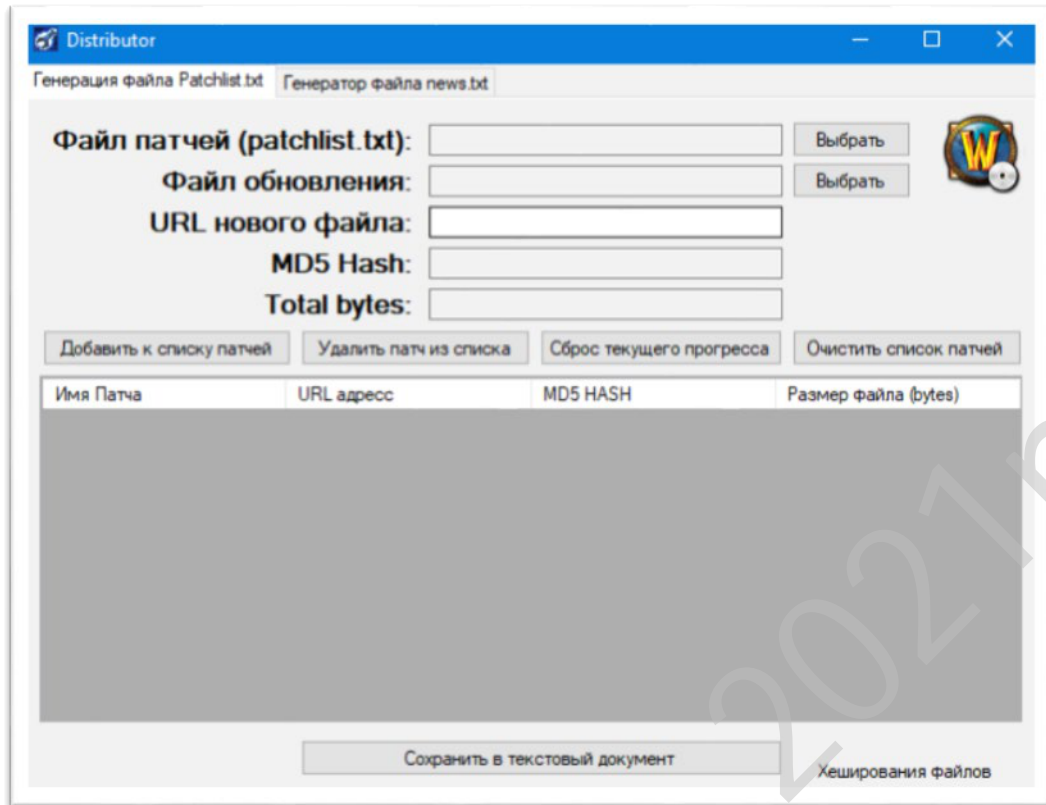


Рисунок 5.11 - Загальний вигляд програми для генерації списку оновлень

Хід створення запису про новому файлі поновлення

- Файл патчів: Треба вибрати існуючий файл patchlist.txt або новостворений.
- Файл поновлення: Це файли .MPQ або патчений Wow.exe. Вибирається з диска на комп'ютері.
- URL нового файлу: Це пряма повна посилання для завантаження з інтернету раніше обраного файлу оновлення.
- MD5 Hash і Total bytes: Це автоматичні розраховуються дані про файлі.
- Якщо ви виконали кроки 1-4, то можете сміливо натискати «Додати до списку патчів», що обнулить значення полів з пунктів 2-4 і додасть в таблицю новий запис. Якщо Ви хочете додати ще файли в список оновлень, то виконайте пункти 2-4, а якщо не хочете нічого більше додавати, то натисніть «Зберегти в текстовий документ».

Додаткові можливості:

- «Видалити патч зі списку» - видаляє активну запис з таблиці.
- «Скидання поточного прогресу» - очищає дані полів з пунктів 2-4.
- «Очистити список патчів» - видалення всіх записів з таблиці.

Поширені помилки:

- Розрахований MD5 хеш файлу не збігається з контрольною сумою завантаження, що призводить до постійної завантаженні файлу з інтернету при кожному запуску програми;
- Файл згенерований і був виправлений вручну, що призводить до неправильної роботи парсера і завантажувача, а також до подальшого фарбую додатки.

Новини

Вкладка «Генератор файлу news.txt».

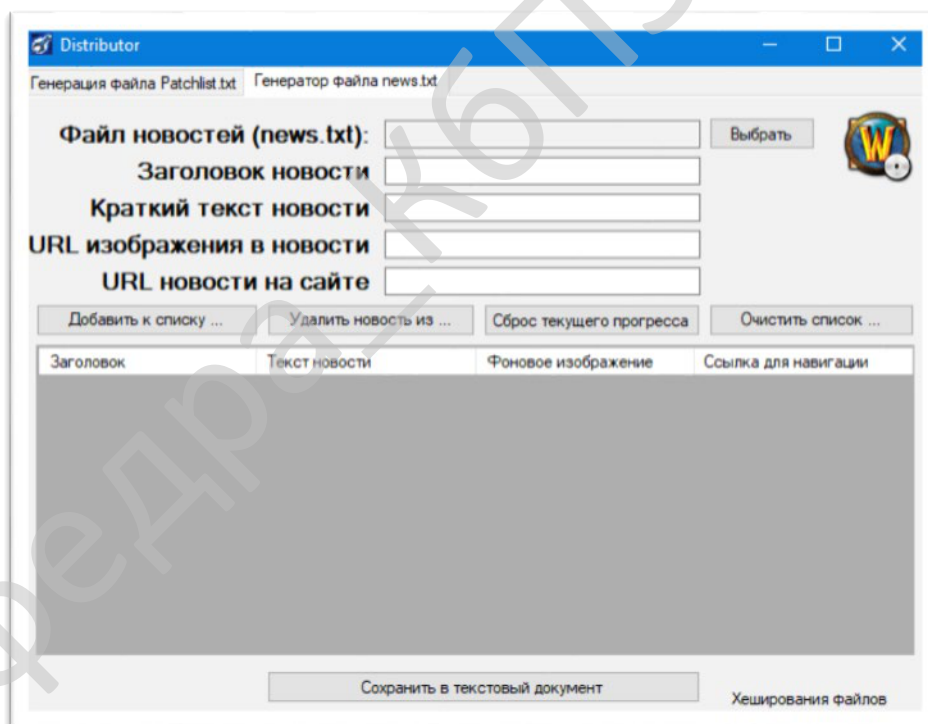


Рисунок 5.12 - Загальний вигляд вкладки генерації новин

Файл новин: Даний пункт аналогічний генерації списку оновлень.

Або треба вибрати існуючий файл з раніше згенерували новинами, або вибираєте новостворений файл.

- «Тема новини»: Тема новини.
- «Короткий текст новини»: Короткий текст новини
- «URL зображення в новини»: повний URL адресу зображення з інтернету, яке в підсумку буде фоновим для новини.
- «URL новини на сайті»: Клацанням по поточній новини відкривається в браузері посилання для більш детального прочитання новини на сайті сервера.
- Решта дії аналогічні тим, що робити при додаванні, збереженні і т.д.

5.4 Методи та обґрунтування тестування програмного продукту

Перед тим як випускати розроблений лаунчер в «вільне плавання» необхідно провести ряд заходів, спрямованих на пошук і виправлення помилок в додатку. Без всього цього не добитися хороших поведінкових показників, а значить проекту складно буде потрапити в топ використовуваних.

Основні види або роботи при тестуванні:

1. Тестування юзабіліті;
2. Тестування продуктивності;
3. Конфігураційне тестування;
4. Функціональне тестування;
5. Тестування сумісності;

Тестування юзабіліті

Будь-який додаток створюється для того, щоб ним скористалися. Зручність використання - важливий якісний показник програми. ІТ індустрія знає безліч прикладів, коли проекти злітали після вдалого виправлення зручності використання. Чим ширше аудиторія, тим важливіше фактор юзабіліті. Тестування юзабіліті включає в себе детальний аналіз поведінки користувачів.

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		59

Для оцінки ергономіки важливо мати дані не тільки про швидкість виконання бізнес-завдання, а й про емоції користувача, міміці особи, тембру голосу.

Тестування продуктивності

В ході етапу тестування продуктивності в першу чергу проводять тестування навантаження, метою якого є перевірка, чи буде система адекватно реагувати на зовнішні впливи в режимі, близькому до режиму реального експлуатації.

Крім навантажувального тестування проводять випробування в умовах мінімальних апаратних засобів і максимального навантаження - стресове тестування, а також, випробування в умовах граничних обсягів оброблюваної інформації - об'ємне тестування.

Виділяють ще один вид тестування: тестування стабільності і надійності, яке включає в себе не тільки тривалий випробування програмного продукту в нормальних умовах, а й здатність його повертатися в нормальний режим функціонування після нетривалих періодів стресових навантажень.

Методи тестування можуть бути різними, однак з самого початку потрібно визначитися зі стратегією і дотримуватися неї - продумати порядок дій і послідовно виконувати їх.

У ході цього тестування я збільшив розмір файлів патчів до 10 Гігабіт. Скачування тривало близько 2 годин, та після того як, закінчилося скачування, я виклав доступ на мій ігровий проект і його скачало 218 користувачів, скарг від користувачів на помилки не було.

Конфігураційне тестування

Тестування додатку (лаунчер) проходило на ОС Windows 7,8,10 (x86, x64 бітності) на Конфігураційне тестування дає впевненість, що додаток запрацює на різних платформах, а значить у максимального числа користувачів. Для ВЕБ-додатків зазвичай вибирають тестування на кросбраузерну. Для Windows додатків - тестування на різних операційних системах і Бітність (x86, x64). Важливою складовою конфігураційного тестування є тестову інфраструктуру:

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		60

для проведення випробувань потрібно постійно підтримувати парк тестових машин. Їх число варіюється від 5 до декількох десятків.

Виявлення програмних помилок є складним завданням. Програмна помилка може не призводити до спостережуваного збою, а, наприклад, породжувати іншу програмну помилку або переводити процес роботи в некоректний стан.

Збій породжується наявністю одного або декількох дефектів - недоліків в компоненті або системі.

Для виявлення програмних помилок використовуються тестові випадки або тести.

Тестовим випадком називають документ, який описує конкретні кроки, умови і параметри, необхідні для аналізу реалізації тесту функції. Кожен тест містить три базові частини.

Передумови - кроки, які переводять систему в стан, придатний для проведення перевірки.

Опис тесту - кроки, які переводять систему зі стану в стан. На підставі отриманого результату робиться висновок про відповідність реалізації заявленим вимогам. Післяумови - кроки, які переводять систему в початкове положення

Функціональне тестування

Один з найбільш часозатратних видів тестування, що вимагає правильні розрахунки функцій. Функціональне тестування є одним з ключових видів тестування, завдання якого - встановити відповідність розробленого програмного забезпечення (ПО) вихідним функціональним вимогам замовника. Тобто проведення функціонального тестування дозволяє перевірити здатність інформаційної системи в певних умовах вирішувати завдання, потрібні користувачам.

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		61

Метод тестування розробленого програмного забезпечення

Для тестування додатку мною було використано мій ігровий проект Horizon-World.net, на якому я провів всі тестування з багатою кількістю користувачів.

Було проведено масу тестувань з різноманітною кількістю файлів різного розміру, які скачувалися через додаток. Були проведені такі тестування:

1. Перевірка надійності заміни файлів на старі (Наприклад файл patch-a.mpq з розміром 6 МБ, був замінений на такий самий файл з розміром 30 МБ без пошкодження клієнту гри.

2. Скачування файлів розміром 5 ГБ з відключенням інтернету за деякий час, та продовження скачування після відновлення інтернет-з'єднання.

3. Автоматична авторизація клієнту

4. Видалення файлів серверу без пошкодження файлів клієнту

5. Оновлення нової версії лаунчера

6. Перевірка працездатності програми на різних версіях Windows, а саме:

- Windows 7 базова, максимальна, професійна (x86-64)
- Windows 8.1 (x86-64)
- Windows 10 (x86-64)

Висновок тестування:

Найкращий вид тестування, це коли є свій проект, на якому є велика кількість користувачів, які будуть користуватися додатком кожен день, та при користуванні помічати якісь недоліки. Як щоб я тестував додаток сам скоріше всього було б витрачено дуже велика кількість часу, а час як розумієте повернути неможливо.

Всі знайдені помилки та відгуки були прийняті на ігровому каналі Horizon World в дискорді.

Протягом всього тестування, яке проходило приблизно три місяці, знайдених критичних помилок знайдено.

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		62

6 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання кваліфікаційної бакалаврської роботи, призначено для гри World of Warcraft.

В представленій кваліфікаційній бакалаврській роботі було розроблено програмне забезпечення ігровий лаунчера гри World of Warcraft.

В ході виконання кваліфікаційної бакалаврської роботи, були вирішені наступні завдання:

- Дослідити та проаналізувати існуючі програмні продукти за напрямком розробки;
- Визначити мову реалізації програмного продукту;
- Розробити план проекту;
- Розробити форми вікон програмного продукту;
- Побудувати структурну схему та виконати її опис;
- Побудувати функціональну схему та виконати її опис;
- Побудувати діаграму процесів системи та виконати її опис;
- Побудувати блок-схему алгоритму основної програми та підпрограм та виконати їх опис;
- Розробити програмний продукт та виконати його тестування;
- Розробити інструкцію користувача та адміністратора.

Розроблене програмне забезпечення має простий, дружній та зручний інтерфейс користувача, що забезпечує легкість у освоєнні роботи програмного продукту, зручність у використанні, і не потребує особливих спеціальних знань.

Програма призначена для виконання під управлінням багатозадачної операційної системи Windows 7/8/10.

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		63

Отже, підбиваючи підсумок, слід сказати, що лаунчер є невід'ємною частиною програм та ігор. Саме завдяки лаунчеру ви можете запускати клієнти, насолоджуватися онлайн-іграми і так далі. Також ви можете персоналізувати свій ігровий проект додавши до нього лаунчер.

Безумовно, найчастіше є можливість запускати додатки і без лаунчеру, але тут є кілька нюансів. Наприклад, вийшла нова версія продукту. Тобто якщо ви запустили гру без лаунчер, то вона буде вважатися оновленою. У деяких випадках це призводить до того, що ви не зможете грати, поки не виконаєте правильний запуск програми.

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		64

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Microsoft. Стили і шаблони [Електронний ресурс] / microsoft // Microsoft. – 2011. – Режим доступу до ресурсу: [https://msdn.microsoft.com/ru-ru/library/vstudio/bb613570\(v=vs.100\).aspx](https://msdn.microsoft.com/ru-ru/library/vstudio/bb613570(v=vs.100).aspx).

2. Professorweb. WPF - Windows Presentation Foundation [Електронний ресурс] / Professorweb // Professorweb. – 2017. – Режим доступу до ресурсу: http://professorweb.ru/my/WPF/base_WPF/level1/info_WPF.php

3. Метью М.Л. WPF / Windows Presentation Foundation в .NET 4.5 с прикладами на C#.

4. . Короткі уроки і готові приклади роботи в WPF [Електронний ресурс] / Christian Moser // WPF. – 2011. – Режим доступу до ресурсу: <http://wpftutorial.net/Home.html>.

5. Еспозіто, Д. Програмування на основі Microsoft ASP.NET MVC. / Д. Еспозіто. - М.: Російська редакція, 2012. - 464 с.

6. Wpf T. The complete WPF tutorial [Електронний ресурс] / Tutorial Wpf // WPF. – 2007. – Режим доступу до ресурсу: <https://www.wpf-tutorial.com>.

7. Понамарьов, В. Програмування на C ++ / C # в Visual Studio.NET 2003 / В. Понамарьов. - СПб.: BHV, 2004. - 352 с.

8. Дмитро Л. Керівництво по WPF [Електронний ресурс] / Лахода Дмитро // Metanit. – 2016. – Режим доступу до ресурсу: <https://metanit.com/sharp/wpf/>.

9. Драч М. Відео курс в WPF [Електронний ресурс] / Максим Драч // Itvdn. – 2017. – Режим доступу до ресурсу: <https://itvdn.com/ru/video/wpf>.

10. Казанський, А.А. Об'єктно-орієнтоване програмування на Visual Basic 2010 і Visual C # 2010 в середовищі розробки Microsoft Visual Studio: Навчальний посібник / А.А. Казанський. - М.: МГСУ, 2012. - 422 с.

11. Гуриков, С.Р. Введення в програмування на мові Visual C #: Уч. сел / С.Р. Гуриков. - М.: Форум, 2013. - 448 с.

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		65

12. Гуріков, С.Р. Введення в програмування на мові Visual C #: Навчальний посібник / С.Р. Гуріков. - М.: Форум, 2011. - 128 с.
13. Котов О. Мова програмування C # 7 і платформи .NET і .NET Core / Олег Котов. – California: Digital project, 2008. – 310 с. – (Ендрю Троелсен).
14. Підготовка розробника за мовою c # на платформі .NET [Електронний ресурс] // Skillcoding. – 2012. – Режим доступу до ресурсу: <http://skillcoding.com/Free.aspx>.
15. Васильєв, О.М. Програмування на C # для початківців. Особливості мови / О.М. Васильєв. - М.: ЕКСМО, 2017. - 128 с.
16. Вагнер, Б. C #. Ефективне програмування. 50 рекомендацій щодо вдосконалення програмування на C # / Б. Вагнер; Пер. з англ. М. Горелик. - М.: Лорі, 2013. - 256 с
17. Майо Дж. Самовчитель Microsoft Visual Studio 2010 року; БХВ-Петербург - М., 2011. - 464 с.
18. Понамарьов В'ячеслав Програмування на C ++ / C # в Visual Studio .NET 2003; БХВ-Петербург - М., 2004. - 352 с.
19. Білліг В. А. Основи об'єктного програмування на C # (C # 3.0, Visual Studio 2008); Інтернет-університет інформаційних технологій, Біном. Лабораторія знань - М., 2010. - 584 с.
20. Албахари Д. C # 7.0. Довідник. Повний опис мови / Джозеф Албахари. – New York: Digital project, 2011. – 240 с. – (Албахари Бен).
21. Уроки C# [Електронний ресурс] // Proger. – 2020. – Режим доступу до ресурсу: <https://itproger.com/course/csharp>.
22. Джеффри Р. JANE AUSTEN - BIOGRAPHY, STYLE, NOVEL OF MANNERS / Рихтер Джеффри. – London: Питер, 2012. – 422 с. – (CLR via).
23. Документація по C # [Електронний ресурс] // Microsoft. – 2020. – Режим доступу до ресурсу: <https://docs.microsoft.com/ru-ru/dotnet/csharp/fundamentals/types/>.

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		66

49. Печерський, А. Мова XML - Практичне введення [Текст] / А. Печерський. - СПб .: Нев. діалект, 2002. - 630с.

50. WPF (WPF .NET) [Електронний ресурс] // Microsoft. – 2021. – Режим доступу до ресурсу: <https://docs.microsoft.com/en-us/dotnet/desktop/wpf/get-started/create-app-visual-studio?view=netdesktop-5.0>.

Кафедра КБПЗ – 2021 рік

					КБР-123.21.0051.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		69

Додаток А
(обов'язковий)

Технічне завдання

ЗМІСТ

1	Найменування та область застосування.....	2
2	Підстава для розробки.....	2
3	Мета та призначення розробки.....	2
4	Джерела розробки.....	2
5	Технічні вимоги.....	2
5.1	Вміст проекту.....	2
5.2	Показники призначення.....	3
5.3	Вимоги до функціональних характеристик.....	3
5.4	Вимоги до архітектури.....	3
5.5	Вимоги до надійності.....	4
5.6	Умови експлуатації.....	4
5.7	Вимоги до складу та параметрів технічних засобів.....	4
5.8	Вимоги до інформаційної і програмної сумісності.....	4
5.8.1	Обладнання.....	5
5.8.2	Мова програмування.....	4
5.8.3	Вхідні дані.....	5
5.8.4	Вихідні дані.....	5
6	Вимоги до програмної документації.....	5
7	Перелік документів, що розробляються.....	5
8	Етапи розробки.....	6
9	Порядок контролю та приймання.....	6

					КБР-123.21.0051.00.00.ТЗ			
Вим.	Арк.	№ документа	Підпис	Дата				
Розробив	Думенко О.А.				Програмне забезпечення ігрового лаунчера комп'ютерної гри <i>World of Warcraft</i>	Літ.	Аркуш	Аркушів
Перевірів	Савеленко О.К.					Б	1	6
Н. Контр.	Гермак В.С.				ЦНТУ КІ-19 2ск			
Затв.	Смірнов О.А.				6			

1 Найменування та область застосування

Це технічне завдання розповсюджується на розробку ігрового завантажувача для гри World of Warcraft.

2 Підстава для розробки

Підставою для розробки служить завдання на кваліфікаційну бакалаврську роботу, видане на кафедрі кібербезпеки та програмного забезпечення (наказ № ____ від _____ 20__ р.).

3 Мета та призначення розробки

Метою кваліфікаційної бакалаврської роботи є підвищення ефективності роботи системи бухгалтерського обліку на основі впровадження нових інформаційних технологій і застосування сучасних засобів програмування.

4 Джерела розробки

Джерелом цієї кваліфікаційної бакалаврської роботи є розробки, які ведуться на ігровому проекті Horizon World.

5 Технічні вимоги

5.1 Вміст проекту

Складовими розробки є:

– аналіз існуючих ігрових завантажувачів на предмет їх функціональних можливостей;

					КБР-123.21.0051.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		6

– вибір і обґрунтування методики побудови додатку і засобів його реалізації;

– розробка структур даних і механізму їхньої взаємодії;

– розробка програми, яка реалізує алгоритми роботи компоненту, а також розрахунок його часових характеристик в рамках додаткового завдання.

5.2 Показники призначення

Система повинна забезпечувати:

– кнопки переходу на веб-ресурси;

– автоматична авторизація на сервері;

– чистка кешу пам'яті гри;

– оновлення та видалення скачуваних оновлень гри;

– автоматичне оновлення програми;

– інформація про час оновлення;

– інформаційне вікно слайдів новин.

5.3 Вимоги до функціональних характеристик

Розроблене програмне забезпечення повинно забезпечувати користувача ігрового серверу, швидким оновлення гри, багатофункціональним та зручним інтерфейсом.

5.4 Вимоги до архітектури

Компонент, що розробляється повинен використовувати архітектуру клієнт/сервер, для скачування файлів для ігрового серверу.

					КБР-123.21.0051.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		6

5.5 Вимоги до надійності

Компонент повинен використати існуючі угоди по стандартним викликам процедур, функцій, засобів і форм, визначених технічною документацією на середовище розробки.

5.6 Умови експлуатації

Автоматизовані робочі місця користувачів системи повинні задовольняти наступним умовам експлуатації:

- температура повітря: 18-22⁰ С;
- відносна вологість повітря при 20⁰ С до 80%;
- атмосферний тиск 107 кПа.

5.7 Вимоги до складу і параметрів технічних засобів

Компонент повинен бути реалізований на ЕОМ типу IBM PC в операційному середовищі Windows 10 і орієнтований на сумісні з цією платформою зовнішні пристрої, мережне обладнання і прикладне програмне забезпечення.

5.8 Вимоги до інформаційної та програмної сумісності

Сумісність програмного забезпечення повинна бути забезпечена працюючи під управлінням ОС Windows 10.

					КБР-123.21.0051.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		6

5.8.1 Обладнання

Ноутбук lenovo ideapad l340 gaming Core i7-9750H® ОЗП 16 ГБ, тип DDR4, Частота оперативної пам'яті 2400 МГц.

5.8.2 Мова програмування

C#.

5.8.3 Вхідні дані

Опис алгоритму роботи запропонованої системи.

5.8.4 Вихідні дані

Робоча програма.

6 Вимоги до програмної документації

Програмна продукція повинна бути представлена в вигляді опису структури даних, схем і описів алгоритмів, інструкції користувача, а також текстів вхідних модулів програмного забезпечення в відповідності з ЄСПД.

7 Перелік документів, які необхідно розробити

- | | |
|--|-------------|
| – Структурна схема системи повна | – 1 аркуш. |
| – Функціональна схема системи | – 1 аркуш. |
| – Діаграма процесів | – 1 аркуш. |
| – Блок-схеми алгоритму роботи програми | – 3 аркуша. |

					КБР-123.21.0051.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		6

8 Етапи розробки

На рівні проекту розробляються (терміни виконання етапів див. в "Завданні на кваліфікаційну бакалаврську роботу"):

8.1 Збір і обробка інформації по темі кваліфікаційної бакалаврської роботи. Постановка задачі на виконання кваліфікаційної бакалаврської роботи (складання ТЗ).

8.2 Проведення досліджень або експериментальних робіт для уточнення основних положень кваліфікаційної бакалаврської роботи.

8.3 Розробка функціональних схем, блок-схем алгоритмів роботи програмного забезпечення компоненту.

8.4 Побудова схем взаємодії структур даних.

8.5 Створення прототипу компоненту. Створення програмного продукту.

8.6 Відлагодження компоненту, аналіз отриманих результатів.

8.7 Оформлення пояснювальної записки і виконання робіт по графічній частині.

9 Порядок контролю і приймання

9.1 Подання кваліфікаційної бакалаврської роботи на попередній захист 22.05.2021 р.

9.2 Подання кваліфікаційної бакалаврської роботи на захист 14.06.2021 р.

					КБР-123.21.0051.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		6

Додаток Б
(обов'язковий)

Міністерство освіти і науки України
Центральноукраїнський національний технічний університет

ЗАТВЕРДЖУЮ

Керівник кваліфікаційної бакалаврської роботи

_____ Савеленко О.К.

*Програмне забезпечення ігрового лаунчера комп'ютерної гри **World of Warcraft***

Лістинг програми

Код документу 12

Носій: CD/DVD-диск

Загальна кількість аркушів: 38

Літера: РП

Кропивницький – 2021 року

```

/// <summary>

// LNewVer
/// </summary>
<Window x:Class="Launcher.LNewVer"
        xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        Title="Обновление инструментария" Height="187" Width="413"
        ResizeMode="NoResize" Topmost="True" WindowStartupLocation="CenterScreen"
        WindowStyle="ToolWindow" Closing="Window_Closing">
    <Grid>
        <RichTextBox AcceptsReturn="False" IsReadOnly="True"
        Margin="10,35,10,51" BorderBrush="White">
            <RichTextBox.Resources>
                <Style TargetType="Hyperlink">
                    <Setter Property="Cursor" Value="Hand" />
                    <EventSetter Event="MouseLeftButtonDown"
                    Handler="Hyperlink_MouseLeftButtonDown" />
                </Style>
            </RichTextBox.Resources>
            <FlowDocument>
                <Paragraph FontSize="20">
                    Доступна новая версия лаунчера. Перейдите по
                    <Hyperlink NavigateUri="">#</Hyperlink>
                    для загрузки.
                </Paragraph>
            </FlowDocument>
        </RichTextBox>
    </Grid>
</Window>

/// <summary>
// LNewVer.xaml.cs
/// </summary>
using System.Windows;
using System.Windows.Documents;
using System.Windows.Input;
using System.Diagnostics;

namespace Launcher
{
    /// <summary>
    // Логика взаимодействия для LNewVer.xaml
    /// </summary>
    public partial class LNewVer
    {
        public LNewVer()
        {
            InitializeComponent();
        }

        private void Hyperlink_MouseLeftButtonDown(object sender,
        MouseEventArgs e)
        {
            var hyperlink = (Hyperlink)sender;
            Process.Start(hyperlink.NavigateUri.ToString());
            Application.Current.Shutdown();
        }

        private void Window_Closing(object sender,
        System.ComponentModel.CancelEventArgs e)
        {
            Application.Current.Shutdown();
        }
    }
}

```

```

    }
  }
}

/// <summary>
// MainWindow
/// </summary>
<Window
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:local="clr-namespace:Launcher" mc:Ignorable="d" x:Name="window"
    x:Class="Launcher.MainWindow"
        Title="World of Warcraft Launcher" Height="650" Width="1024"
    MinWidth="800" MinHeight="500" ResizeMode="CanResizeWithGrip"
    WindowStartupLocation="CenterScreen" Icon="img/101.ico" BorderThickness="0"
    Background="{x:Null}" IsEnabled="False" WindowStyle="None"
    AllowsTransparency="True">
    <Window.Resources>
        <ZoomPercentageConverter x:Key="ZoomPercentageConverter"/>
    </Window.Resources>
    <Window.TaskbarItemInfo>
        <TaskbarItemInfo x:Name="TaskbarProgress" ThumbnailClipMargin="2"
        ProgressState="None">
            <TaskbarItemInfo.ThumbButtonInfos>
                <ThumbButtonInfo x:Name="TaskbarPlay" IsEnabled="False"
                Description="Играть" ImageSource="img/play-thumb.png"
                Click="TaskbarPlay_Click"/>
            </TaskbarItemInfo.ThumbButtonInfos>
        </TaskbarItemInfo>
    </Window.TaskbarItemInfo>
    <Grid x:Name="MainGrid" MouseDown="Grid_MouseDown" ClipToBounds="True"
    Margin="0">
        <Grid Background="#FF260A23">
            <Canvas Background="#FF311102" Height="800" Width="1920"
            VerticalAlignment="Top" HorizontalAlignment="Center" Margin="-510,0,-510,-
            135">
                <Image Source="img/main.jpg" Stretch="UniformToFill"
                StretchDirection="UpOnly"/>
                <Border Height="80" BorderBrush="#33FFFFFF"
                BorderThickness="0,0,0,1">
                    <Image Source="img/main.jpg" Stretch="UniformToFill"
                    StretchDirection="UpOnly" Margin="0,0,0,0">
                        <Image.Effect>
                            <BlurEffect Radius="25"/>
                        </Image.Effect>
                    </Image>
                </Border>
            </Canvas>
            <Rectangle Stroke="#33FFFFFF"/>
            <Grid Cursor="{Binding Cursor, ElementName=MainGrid}">
                <Grid.ColumnDefinitions>
                    <ColumnDefinition Width="3*" />
                    <ColumnDefinition Width="5*" />
                </Grid.ColumnDefinitions>
                <Grid.RowDefinitions>
                    <RowDefinition Height="77" />
                    <RowDefinition />
                    <RowDefinition Height="100" />
                </Grid.RowDefinitions>

```

```

        <Rectangle Grid.Row="2" StrokeThickness="0" Margin="1"
Grid.ColumnSpan="2">
        <Rectangle.Fill>
        <LinearGradientBrush EndPoint="0.5,1"
StartPoint="0.5,0">
        <GradientStop Color="#FF311102" Offset="1"/>
        <GradientStop Color="#00311102" Offset="0.267"/>
        <GradientStop Color="#99311102"/>
        <GradientStop Color="#00311102" Offset="0.388"/>
        </LinearGradientBrush>
        </Rectangle.Fill>
    </Rectangle>

    <Button x:Name="btn_close" Cursor="{Binding Cursor,
ElementName=version}" Margin="0,5,5,0" Style="{DynamicResource
SimpleButton}" Click="btn_close_Click" FontSize="18" ToolTip="Закреть
приложение" HorizontalAlignment="Right" Width="24" Height="24"
VerticalAlignment="Top" BorderThickness="0" FontFamily="Arial Rounded MT"
IsTabStop="False" Grid.Column="1" >
        <Path
            Width="16"
            Height="16"
            SnapsToDevicePixels="False"
            StrokeThickness="3"
            Data="M 0 0 L 7 7 M 0 7 L 7 0"
Stretch="Fill" Stroke="White"/>
    </Button>
    <Button x:Name="btn_min" Cursor="{Binding Cursor,
ElementName=version}" Margin="0,5,34,0" Style="{DynamicResource
SimpleButton}" FontSize="24" ToolTip="Свернуть приложение"
HorizontalAlignment="Right" Width="24" Height="24" VerticalAlignment="Top"
BorderThickness="0" FontFamily="Arial Rounded MT" IsTabStop="False"
Click="btn_min_Click" Padding="1" VerticalContentAlignment="Center"
Foreground="White" Grid.Column="1" >
        <Path
            Width="16"
            Height="16"
            SnapsToDevicePixels="False"
            StrokeThickness="3"
            Data="M 0 0 L 7 0" Stretch="Fill"
Stroke="White" VerticalAlignment="Bottom"/>
    </Button>
    <Button x:Name="btn_settings" Cursor="{Binding Cursor,
ElementName=version}" Margin="0,5,63,0" Style="{DynamicResource
SimpleButton}" FontSize="16" ToolTip="Настройка приложения"
HorizontalAlignment="Right" Width="24" Height="24" VerticalAlignment="Top"
BorderThickness="0" FontFamily="Arial Rounded MT" IsTabStop="False"
Padding="1" VerticalContentAlignment="Center" Click="btn_settings_Click"
Grid.Column="1">
        <Path
            Width="16"
            Height="16"
            SnapsToDevicePixels="False"
            StrokeThickness="0"
            Data="F1 M 38,23.5C 38.8643,23.5
39.7109,23.5756 40.5337,23.7206L 42.6275,18.5381L 48.1901,20.787L
46.0964,25.9692C 47.6473,27.0149 48.9851,28.3527 50.0308,29.9036L
55.213,27.8099L 57.4619,33.3725L 52.2794,35.4664C 52.4244,36.2891
52.5,37.1357 52.5,38C 52.5,38.8643 52.4244,39.7109 52.2794,40.5337L
57.4619,42.6275L 55.213,48.1901L 50.0308,46.0964C 49.0795,47.5073
47.8865,48.7418 46.5112,49.7405L 48.7844,54.8462L 43.3041,57.2891L
41.0307,52.1828C 40.0533,52.3906 39.0394,52.5 38,52.5C 37.1357,52.5
36.2891,52.4244 35.4664,52.2794L 33.3725,57.462L 27.8099,55.213L

```

```

29.9036,50.0309C 28.3527,48.9851 27.0149,47.6473 25.9691,46.0964L
20.787,48.1901L 18.538,42.6275L 23.7206,40.5336C 23.5756,39.7109 23.5,38.8643
23.5,38C 23.5,37.1357 23.5756,36.2891 23.7206,35.4664L 18.538,33.3725L
20.787,27.8099L 25.9691,29.9036C 26.9205,28.4927 28.1135,27.2582
29.4889,26.2594L 27.2157,21.1537L 32.6959,18.7109L 34.9694,23.8172C
35.9468,23.6094 36.9606,23.5 38,23.5 Z M 38,28C 32.4771,28 28,32.4772 28,38C
28,43.5229 32.4771,48 38,48C 43.5228,48 48,43.5229 48,38C 48,32.4772
43.5228,28 38,28 Z " Stretch="Fill" Fill="White"/>
</Button>
<Label x:Name="version" Foreground="#FFF9EB8F" Content=""
Margin="0,0,92,0" FontSize="13" Cursor="Hand" MouseDown="Version_MouseDown"
ToolTip="Просмотреть историю обновлений программы" Height="33"
VerticalAlignment="Top" VerticalContentAlignment="Center"
RenderTransformOrigin="0.5,0.5" HorizontalContentAlignment="Right"
HorizontalAlignment="Right" Width="90" Grid.Column="1"/>
<local:NewsLoaderControl x:Name="news_box"
Margin="170,10,30,10" Grid.Row="1" Foreground="White"
Loaded="news_box_Loaded" Grid.ColumnSpan="2"/>
<Label x:Name="contentNewsLoadIndicator" Grid.Row="1"
Margin="1,0,1,0" Foreground="White" HorizontalContentAlignment="Center"
VerticalContentAlignment="Center" Visibility="Hidden" Grid.ColumnSpan="2">
<TextBlock x:Name="newsLoadIndicator" TextWrapping="Wrap"
FontSize="12pt"><Run Text="Загрузка новостей..."></TextBlock>
</Label>
<RichTextBox x:Name="NewsBox" Margin="1,0" Grid.Row="1"
AcceptsReturn="False" IsReadOnly="True" ScrollViewer.CanContentScroll="True"
VerticalScrollBarVisibility="Auto" Foreground="White" Focusable="False"
IsTabStop="False" Template="{DynamicResource NewsBox}" Cursor="Arrow"
Padding="10" FontSize="15" BorderThickness="0,0,0,1" Background="#99171E25"
BorderBrush="#33FFFFFF" Visibility="Hidden" Grid.ColumnSpan="2" >
<RichTextBox.Resources>
<!--<Style TargetType="Hyperlink">
<Setter Property="Cursor" Value="Hand" />
<Setter Property="Foreground" Value="#FF363636"/>
<EventSetter Event="MouseLeftButtonDown"
Handler="Hyperlink_MouseLeftButtonDown" />
</Style>-->
<Style TargetType="{x:Type Hyperlink}">
<Setter Property="Foreground" Value="#FF00aeff"/>
<Setter Property="TextDecorations" Value="None"/>
<Setter Property="Cursor"
Value="pack://application:,,,/img/cursors/WOW-ENLACE-CURSOR.cur"/>
<EventSetter
Event="ContentElement.MouseLeftButtonDown"
Handler="Hyperlink_MouseLeftButtonDown" />
<Style.Triggers>
<Trigger Property="IsMouseOver" Value="True">
<Setter Property="Foreground"
Value="#ffffffff"/>
</Trigger>
<!--<Trigger Property="IsEnabled"
Value="False">
<Setter Property="Foreground"
Value="{DynamicResource {x:Static SystemColors.GrayTextBrushKey}}"/>
</Trigger>-->
</Style.Triggers>
</Style>
</RichTextBox.Resources>
<FlowDocument/>
</RichTextBox>

```

```

        <Grid Cursor="{Binding Cursor, ElementName=MainGrid}"
x:Name="DownloadBar" Grid.Row="2" Margin="213,0,10,10" Height="75"
VerticalAlignment="Bottom" Visibility="Hidden" Grid.ColumnSpan="2">
    <Grid.RowDefinitions>
        <RowDefinition Height="23*" />
        <RowDefinition Height="19*" />
        <RowDefinition Height="19*" />
    </Grid.RowDefinitions>
    <Label x:Name="labelprogress" Content=""
Margin="17,0,11,1" Foreground="#FFE8E8E8" Background="#00836359" Padding="0"
Height="19" VerticalAlignment="Bottom" />
    <Grid x:Name="totalProgressGrid" Grid.Row="1">
        <ProgressBar x:Name="progress" Margin="16,0,76,0"
Style="{DynamicResource ProgressBar}" ToolTip="Общий прогресс обновления"
Height="19" VerticalAlignment="Bottom">
            <ProgressBar.Foreground>
                <LinearGradientBrush EndPoint="0.5,1"
StartPoint="0.5,0">
                    <GradientStop Color="#FF4292A7"
Offset="0" />
                    <GradientStop Color="#FF3A8DA1"
Offset="0.5" />
                    <GradientStop Color="#FF347F95"
Offset="0.5" />
                    <GradientStop Color="#FF307B91"
Offset="1" />
                </LinearGradientBrush>
            </ProgressBar.Foreground>
            <ProgressBar>
                <Label Content="{Binding Value, ElementName=progress,
StringFormat=\{0:0\}}" Margin="0,0,35,0" Foreground="#7FFFFFFF"
FontSize="15" HorizontalAlignment="Right" Width="36"
HorizontalContentAlignment="Right" FontFamily="Tahoma" ToolTip="Общий
прогресс обновления" Padding="0" VerticalContentAlignment="Bottom" />
                <Label Content="%" Foreground="#7FFFFFFF"
FontSize="15" HorizontalAlignment="Right" Width="25"
HorizontalContentAlignment="Center" Margin="0,0,10,0" FontFamily="Tahoma"
ToolTip="Общий прогресс обновления" Padding="0"
VerticalContentAlignment="Bottom" />
            </Grid>
        </Grid>
        <Grid x:Name="currentProgressGrid" Grid.Row="2">
            <ProgressBar x:Name="progress_file"
Margin="16,0,76,0" Style="{DynamicResource ProgressBar}" ToolTip="Текущий
прогресс обновления" Height="19" VerticalAlignment="Bottom">
                <ProgressBar.Foreground>
                    <LinearGradientBrush EndPoint="0.5,1"
StartPoint="0.5,0">
                        <GradientStop Color="#FF4292A7"
Offset="0" />
                        <GradientStop Color="#FF3A8DA1"
Offset="0.5" />
                        <GradientStop Color="#FF347F95"
Offset="0.5" />
                        <GradientStop Color="#FF307B91"
Offset="1" />
                    </LinearGradientBrush>
                </ProgressBar.Foreground>
            </ProgressBar>
            <Label Content="{Binding Value,
ElementName=progress_file, StringFormat=\{0:0\}}" Margin="0,0,35,0"
Foreground="#7FFFFFFF" FontSize="15" HorizontalAlignment="Right" Width="36"
HorizontalContentAlignment="Right" FontFamily="Tahoma" ToolTip="Текущий
прогресс обновления" Padding="0" VerticalContentAlignment="Bottom" />
        </Grid>
    </Grid>

```

```

        <Label Content="%" Foreground="#7FFFFFFF"
FontSize="15" HorizontalAlignment="Right" Width="25"
HorizontalContentAlignment="Center" Margin="0,0,10,0" FontFamily="Tahoma"
ToolTip="Текущий прогресс обновления" Padding="0"
VerticalContentAlignment="Bottom"/>

    </Grid>
</Grid>
<Label x:Name="labelmsg" Grid.Row="2"
Content="Инициализация..." Margin="10,0,0,82" Foreground="#7FEAEAEA"
HorizontalAlignment="Left" Width="296" Height="14" VerticalAlignment="Bottom"
Padding="0"/>
    <Button x:Name="btn_play" Grid.Row="2" Margin="10,0,0,10"
Click="btn_play_Click" Height="67" VerticalAlignment="Bottom"
HorizontalAlignment="Left" Width="180" IsTabStop="False" Content="ИГРАТЬ"
Style="{DynamicResource PlayButton}" FontFamily="Verdana" FontSize="26.667"
Cursor="{Binding Cursor, ElementName=version}" BorderThickness="0"/>
    <Button x:Name="link_main" Content="ГЛАВНАЯ"
HorizontalAlignment="Left" Margin="24,5,0,470" Style="{DynamicResource
LinkButton}" Width="115" FontSize="16" Click="link_main_Click"
Grid.RowSpan="2" RenderTransformOrigin="1.878,0.407"/>
    <Button x:Name="link_cabinet" Content="ФОРУМ"
HorizontalAlignment="Left" Margin="24,3,0,391" Style="{DynamicResource
LinkButton}" Width="115" FontSize="16" Click="link_cabinet_Click"
Grid.Row="1"/>
    <Button x:Name="link_registration" Content="РЕГИСТРАЦИЯ"
HorizontalAlignment="Left" Margin="4,87,0,308" Style="{DynamicResource
LinkButton}" Width="150" FontSize="16" Click="link_registration_Click"
RenderTransformOrigin="-1.14,2.195" Grid.Row="1"/>
    <Button x:Name="link_social" Content="VKONTAKTE"
HorizontalAlignment="Left" Margin="10,165,0,229" Style="{DynamicResource
LinkButton}" Width="144" FontSize="16" Click="link_social_Click"
Grid.Row="1"/>
    <Button x:Name="link_social_Copy"
Content="ПРОЕКТ&#xA;МАЙНКРАФТ" HorizontalAlignment="Left"
Margin="179,2,0,469" Style="{DynamicResource LinkButton}" Width="130"
FontSize="16" Click="link_social_Copy_Click" Grid.RowSpan="2"/>
    <Button x:Name="link_social_Copy1" Content="DISCORD"
HorizontalAlignment="Left" Margin="10,244,0,150" Style="{DynamicResource
LinkButton}" Width="144" FontSize="16" Click="link_social_Copy1_Click"
Grid.Row="1"/>
    </Grid>
</Grid>
</Grid>
</Window>

/// <summary>
// MainWindow.xaml.cs
/// </summary>
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Diagnostics;
using System.Drawing;
using System.Globalization;
using System.IO;
using System.Linq;
using System.Net;
using System.Reflection;
using System.Runtime.InteropServices;
using System.Security.Cryptography;

```

```

using System.Text;
using System.Threading;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Documents;
using System.Windows.Forms;
using System.Windows.Input;
using System.Windows.Media.Effects;
using System.Windows.Shell;
using Launcher.HelpClasses;
using Application = System.Windows.Application;
using Cursor = System.Windows.Input.Cursor;
using Cursors = System.Windows.Input.Cursors;
using MessageBox = System.Windows.MessageBox;
using MouseEventArgs = System.Windows.Input.MouseEventArgs;

namespace Launcher
{
    /// <summary>
    /// Логіка взаємодії для MainWindow.xaml
    /// </summary>
    public partial class MainWindow
    {
        #region Variables

        #region dll's;

        [DllImport("user32.dll")]
        private static extern IntPtr SendMessage(IntPtr hWnd, uint msg,
        IntPtr wParam, IntPtr lParam);

        private const uint WmKeydown = 0x0100;
        private const uint WmKeyUp = 0x0101;
        private const uint WmChar = 0x0102;
        private const int VkReturn = 0x0D;
        private const int VkTab = 0x09;

        #endregion dll's;

        #region const;
        private delegate void UpdateProgress(int percent, long bytesReceived,
        long totalBytesReceive);
        private delegate void MakeVisibleInvisible(bool visible);

        private readonly Queue<PatchFileInfo> _patchQueue = new
        Queue<PatchFileInfo>();

        private readonly string _tempPath = Path.GetTempFileName();
        private static readonly string AppRoamingPath = Path.Combine(
        Environment.GetFolderPath(Environment.SpecialFolder.ApplicationData),
        Assembly.GetEntryAssembly()?.GetName().Name ?? "Launcher");
        private readonly string _updatePath = Path.Combine(AppRoamingPath,
        "UpdatedFiles.json");

        public Stopwatch StopWatch = new Stopwatch();

        private Dictionary<string, string> _filesCompleted = new
        Dictionary<string, string>();
        private int _count, _length;
        private string _gPath, _pListUri, _pListDel;
        public bool AnyDownloads;

```

```

private long _totalBytes, _currentBytes, _currentFileBytes;
public long CurrentBytes2, CurrentFileBytes2;

#endregion const;

#endregion

/// <summary>
/// Patch file class
/// </summary>
private class PatchFileInfo
{
    public string Url { get; }
    public string Name { get; }
    public string File { get; }
    public string Md5Hash { get; }
    public long FileBytes { get; }

    /// <summary>
    /// Class initializer
    /// </summary>
    /// <param name="url">Direct download url</param>
    /// <param name="name">File name</param>
    /// <param name="file">Where to download file</param>
    /// <param name="fileBytes">File size in bytes</param>
    /// <param name="md5">MD5 hash</param>
    public PatchFileInfo(string url, string name, string file, long
fileBytes, string md5)
    {
        Url = url;
        Name = name;
        Name = name;
        File = file;
        FileBytes = fileBytes;
        Md5Hash = md5;
    }
}

public MainWindow()
{
    InitializeComponent();

    var normalCursor = Application.GetResourceStream(new
Uri("pack://application:,,,/img/cursors/wow.cur"))?.Stream;
    var readlCursor = Application.GetResourceStream(new
Uri("pack://application:,,,/img/cursors/WOW-ESCRIVIR.cur"))?.Stream;
    var hightlCursor = Application.GetResourceStream(new
Uri("pack://application:,,,/img/cursors/WOW-ENLACE-CURSOR.cur"))?.Stream;

    MainGrid.Cursor = new Cursor(normalCursor ?? throw new
NullReferenceException("He найден pecyc wow.cur"));
    version.Cursor = new Cursor(hightlCursor ?? throw new
NullReferenceException("He найден pecyc WOW-ESCRIVIR.cur"));
    NewsBox.Cursor = new Cursor(readlCursor ?? throw new
NullReferenceException("He найден pecyc WOW-ENLACE-CURSOR.cur"));

    SetProgressType(Properties.Settings.Default.ProgressBarType);

    Directory.CreateDirectory(AppRoamingPath);
    if (Utilities.Network.IsInternetConnectionAvailable())
    {
        FileSync();
    }
}

```

```

else
{
    MessageBox.Show("Невозможно подключиться к сети интернет,
    проверьте ваше подключение и повторите попытку", "Ошибка подключения",
    MessageBoxButtons.OK, MessageBoxIcon.Error);
    Application.Current.Shutdown();
}
}

/// <summary>
/// Почати синхронізацію файлу за папкою гри (якщо вона була
встановлена раніше)
/// </summary>
private void FileSync()
{
    var storedPath = Properties.Settings.Default.GameFolder;
    if (File.Exists(Path.Combine(storedPath, "Wow.exe")))
    {
        UpdateClient(storedPath);
        return;
    }

    var localPath = AppDomain.CurrentDomain.BaseDirectory;
    if (File.Exists(Path.Combine(localPath, "Wow.exe")))
    {
        UpdateClient(localPath);
        return;
    }

    var result = MessageBox.Show("Файл \"Wow.exe\" не
    найден!\nПожалуйста поместите программу в папку с игрой или укажите путь к
    папке с игрой!\n\nУказать путь сейчас?", "Ошибка местоположения",
    MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    TryToFindFolder(result);
}

/// <summary>
/// Показати провідник папок, щоб вибрати шлях гри
/// </summary>
private void ShowFolderDialog()
{
    var folder = new FolderBrowserDialog
    {
        Description = @"Выберите папку с клиентом игры",
        RootFolder = Environment.SpecialFolder.MyComputer,
        ShowNewFolderButton = false
    };
    var result = folder.ShowDialog();

    if (result != System.Windows.Forms.DialogResult.OK)
    {
        Application.Current.Shutdown();
        return;
    }

    if (File.Exists(Path.Combine(folder.SelectedPath, "Wow.exe")))
    {
        var folderPath = folder.SelectedPath;
        Properties.Settings.Default.GameFolder = folderPath;
        Properties.Settings.Default.Save();
        FileSync();
        return;
    }
}

```

```

        var retryResult = MessageBox.Show("В выбранной папке файл
        \"Wow.exe\" не найден!\nПожалуйста выберите корректную папку с
        игрой!\n\nПовторить попытку выбора?", "Ошибка выбора папки",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question);
        TryToFindFolder(retryResult);
    }

    /// <summary>
    /// Дозволить користувачеві вибрати папку гри, якщо програма запуску
    не в кореневій папці гри
    /// </summary>
    /// <param name="retryResult">MessageBox result for </param>
    private void TryToFindFolder(MessageBoxResult retryResult)
    {
        switch (retryResult)
        {
            case MessageBoxResult.Yes:
                ShowFolderDialog();
                break;
            case MessageBoxResult.No:
                Application.Current.Shutdown();
                break;
        }
    }

    /// <summary>
    /// ВИЗНАЧИТЬ ВЕРСИЮ КЛІЄНТА І ВСТАНОВИТИ СПИСОК ОНОВЛЕННЯ, РЕАЛІМІСТ,
    ВИДАЛИТИ КЕШ
    /// </summary>
    /// <param name="gPath">Wow game root folder</param>
    private void UpdateClient(string gPath)
    {
        var clientVersion =
        FileVersionInfo.GetVersionInfo(Path.Combine(gPath, "Wow.exe"));
        var v = clientVersion.FileVersion.Split(char.Parse(","));

        // виявлення версії клієнта. виберіть потрібні файли виправлення
        для завантаження та видалення
        switch (v[3].Trim()) {
            case "12340":
                _pListUri = Properties.Settings.Default.PatchDownloadURL;
                _pListDel = Properties.Settings.Default.PatchToDelete;
                break;

            //case " another_client_build_version":

            //Create another string properties in project properties
            // _pListUri =
            Properties.Settings.Default.PatchDownloadURL;
            // _pListDel = Properties.Settings.Default.PatchToDelete;
            default:
                //TODO: CHANGE SERVER NAME AND CLIENT VERSION
                var result = MessageBox.Show("Для игры на сервере
                %Horizon% требуется клиент версии 3.3.5.12340! Поместите программу в
                корректную папку с игрой или укажите путь к папке!\n\nуказать путь сейчас?",
                "Ошибка версии клиента", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
                TryToFindFolder(result);
                break;
        }

        var rPath = Path.Combine(gPath, @"Data\ruRU\realmist.wtf");
    }

```

```

var cPath = Path.Combine(gPath, "Cache");
_gPath = gPath;

try
{
    Directory.Delete(cPath, true);
}
catch (Exception)
{
    // ignored
}

try
{
    var request = (HttpWebRequest)WebRequest.Create(new
Uri(Properties.Settings.Default.RealmListURL));
    var response = (HttpWebResponse)request.GetResponse();
    var sr = new StreamReader(response.GetResponseStream()) ??
throw new InvalidOperationException($"Ошибка получения ответа от
{Properties.Settings.Default.RealmListURL}");
    var realmList = sr.ReadToEnd();

    var attributes = File.GetAttributes(rPath);
    if ((attributes & FileAttributes.ReadOnly) ==
FileAttributes.ReadOnly)
    {
        attributes =
Utilities.File.RemoveAttribute(attributes, FileAttributes.ReadOnly);
        File.SetAttributes(rPath, attributes);
    }

    //TODO: ЗАКОМЕНТУВАТИ ЯКЩО КЛІЄНТ НЕ МАЄ ФАЙЛУ РЕАЛМЛІСТУ
#region <= 3.3.5 realmList changer
using (var writer = new StreamWriter(rPath))
{
    writer.WriteLine(realmList);
}
#endregion

//TODO: РОЗКОМЕНТУВАТИ НАСТУПНУ ЧАСТИНУ КОДУ, ЯКЩО КЛІЄНТ НЕ
МАЄ РЕАЛМЛІСТУ
#region >= 3.3.5 realmList changer
/*
var builder = new StringBuilder();

using (var reader = new StreamReader(rPath))
{
    string line;
    while ((line = reader.ReadLine()) != null)
        builder.AppendLine(line.ToLower().Contains("set
realmList") ? realmList : line);
}

using (var writer = new StreamWriter(rPath))
{
    writer.Write(builder.ToString());
}

*/
#endregion

DeleteOldPatches();

```

```

    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message + "\n\n" + ex.InnerException,
ex.Source);
        Application.Current.Shutdown();
    }
}

/// <summary>
/// Завантажит новини в NewsLoader Control
/// </summary>
private void LoadNews()
{
news_box.SetNews(Properties.Settings.Default.LauncherNewsFileUrl);

        AutoUpdate();
        IsEnabled = true;
    }

    /// <summary>
    /// Видалення старих виправлень або старі можливі оновлення
    /// </summary>
    private void DeleteOldPatches()
    {
        var request = (HttpWebRequest)WebRequest.Create(new
Uri(_pListDel));
        var response = (HttpWebResponse)request.GetResponse();

        var reader = new StreamReader(response.GetResponseStream()) ??
throw new InvalidOperationException($"Ошибка получения ответа от
{_pListDel}");

        string line;

        while ((line = reader.ReadLine()) != null)
        {
            var path = Utilities.Updater.GetPath(_gPath, line);

            try
            {
                File.Delete(path);
            }
            catch (Exception)
            {
                // ignored
            }
        }

        LoadNews();
    }

    private void SaveCurrentCompletedFiles()
    {
        using (var file = File.CreateText(_updatePath))
        {
            file.WriteLine(JsonSerializer<Dictionary<string,
string>>.ToString(_filesCompleted));
        }
    }

    /// <summary>

```

```

/// Starts get update list for self or game client
/// </summary>
private void AutoUpdate()
{
    var request = (HttpWebRequest)WebRequest.Create(new
Uri(Properties.Settings.Default.LauncherVersionUrl));
    var response = (HttpWebResponse)request.GetResponse();
    var sr = new StreamReader(response.GetResponseStream()) ?? throw
new InvalidOperationException($"Ошибка получения ответа от
{Properties.Settings.Default.LauncherVersionUrl}");
    var newVersion = sr.ReadToEnd();

    var assembly = Assembly.GetExecutingAssembly();
    var assemblyName = assembly.GetName();
    var ver = assemblyName.Version;

    var currentVersion = ver.ToString();
    version.Content = $"ver. {currentVersion}";

    DownloadBar.Visibility = Visibility.Hidden;

    if (newVersion.Contains(currentVersion))
    {
        if (Properties.Settings.Default.PatchDownloadURL ==
string.Empty) return;

        var startDownloadBackgroundWorker = new BackgroundWorker();
        startDownloadBackgroundWorker.DoWork +=
startDownloadBackgroundWorker_DoWork;

        //DownloadBar.Visibility = Visibility.Visible;
        btn_play.IsEnabled = false;
        TaskbarPlay.IsEnabled = false;
        progress.Value = 0;
        labelmsg.Content = "Инициализация...";
        _count = 0;
        startDownloadBackgroundWorker.RunWorkerAsync();
    }
    else
    {
        var launcherNewVersion = new LNewVer();
        launcherNewVersion.ShowDialog();
    }
}

private void startDownloadBackgroundWorker_DoWork(object sender,
DoWorkEventArgs e)
{
    var client = new WebClient();
    client.DownloadFileCompleted += client_DownloadFileCompleted;
    client.DownloadFileAsync(new Uri(_pListUri), _tempPath);
    Stopwatch.Start();
}

private void client_DownloadFileCompleted(object sender,
AsyncCompletedEventArgs e)
{
    _length = File.ReadAllLines(_tempPath).Length;

    Console.WriteLine($"-----");
    Console.WriteLine($"Start initialization in
{DateTime.Now:HH:mm:ss.ffffff}.");
    Console.WriteLine($"-----");
}

```

```

// Get map of downloaded files
if (File.Exists(_updatePath))
{
    _filesCompleted = JsonSerializer<Dictionary<string,
string>>.FromString(
        File.ReadAllText(_updatePath));
}

using (var reader = new StreamReader(_tempPath))
{
    string line;

    while ((line = reader.ReadLine()) != null)
    {
        var ex = line.Split(char.Parse("#"));
        var path = Utilities.Updater.GetPath(_gPath, ex[1]);
        var pfi = new PatchFileInfo(ex[0], ex[1], path,
Convert.ToInt64(ex[3]), ex[2]);

        Console.WriteLine($"({_count + 1:000}/{_length:000}]
Start checking {pfi.Name}.");

        if (File.Exists(path))
        {
            if (_filesCompleted.TryGetValue(pfi.Name, out var
hash) && hash == pfi.Md5Hash)
            {
                // Ignore downloaded file
                _count++;
                Console.WriteLine($"{{pfi.Name}} is updated.
{{_length - _count}} more left.");
                continue;
            }

            var fs = new FileStream(path, FileMode.Open,
FileAccess.Read, FileShare.Read);
            MD5 md5 = new MD5CryptoServiceProvider();
            var retVal = md5.ComputeHash(fs);
            fs.Close();
            var sb = new StringBuilder();
            foreach (var b in retVal)
            {
                sb.Append($"{{b:X2}}");
            }

            if (pfi.Md5Hash == sb.ToString())
            {
                // Ignore matched file and add to completed map
                _filesCompleted[pfi.Name] = pfi.Md5Hash;
                _count++;
                Console.WriteLine($"Done with {pfi.Name}.
{{_length - _count}} more left.");
                SaveCurrentCompletedFiles();
                continue;
            }
        }

        _filesCompleted.Remove(pfi.Name);
        _patchQueue.Enqueue(pfi);
        _totalBytes += pfi.FileBytes;
    }
}

```

```

    }

    Console.WriteLine($"-----");
    Console.WriteLine($"End initialization in
{DateTime.Now:HH:mm:ss.ffffff}.");
    Console.WriteLine($"-----");

    while (_patchQueue.Count != 0)
    {
        Stopwatch.Start();
        AnyDownloads = true;
        Dispatcher.Invoke(new
MakeVisibleInvisible(DownloadCompleted), true);

        var pfi = _patchQueue.Dequeue();
        var append = false;
        var currentFile = $"{pfi.File}.{pfi.Md5Hash}.upd";

        var httpReq = (HttpWebRequest)WebRequest.Create(pfi.Url);

        if (File.Exists(currentFile))
        {
            var destinationFileInfo = new FileInfo(currentFile);
            var existedLength = destinationFileInfo.Length;

            _currentFileBytes += existedLength;
            _currentBytes += existedLength;
            httpReq.AddRange((int)existedLength);
            append = true;
        }

        try
        {
            var httpRes = (HttpWebResponse)httpReq.GetResponse();
            var resStream = httpRes.GetResponseStream();

            using (var file = append ? new FileStream(currentFile,
FileMode.Append, FileAccess.Write, FileShare.ReadWrite) : new
FileStream(currentFile, FileMode.Create, FileAccess.Write,
FileShare.ReadWrite))
            {
                const int bufferSize = 1024 * 4;
                var buffer = new byte[bufferSize];
                int bytesReceived;

                while (resStream != null && (bytesReceived =
resStream.Read(buffer, 0, buffer.Length)) != 0)
                {
                    file.Write(buffer, 0, bytesReceived);

                    try
                    {
                        Dispatcher.Invoke(new
UpdateProgress(UpdateProgressbar), new object[] { 0, bytesReceived,
Convert.ToInt32(pfi.FileBytes) });
                    }

                    if
(Properties.Settings.Default.DownloadSpeedLimit > 0)
                    {
                        if (CurrentFileBytes2 * 1000L /
StopWatch.Elapsed.TotalMilliseconds >
Properties.Settings.Default.DownloadSpeedLimit)
                        {

```



```

CurrentBytes2 += bytesReceived;
_currentFileBytes += bytesReceived;
CurrentFileBytes2 += bytesReceived;

int percentTotal =
Convert.ToInt16((Convert.ToDouble(double.Parse(_currentBytes.ToString(Culture
Info.InvariantCulture))) / 1024 / 1024) /
(Convert.ToDouble(double.Parse(_totalBytes.ToString(CultureInfo.InvariantCult
ure))) / 1024 / 1024 / 100));
percent =
Convert.ToInt16((Convert.ToDouble(double.Parse(_currentFileBytes.ToString(Cul
tureInfo.InvariantCulture))) / 1024 / 1024) /
(Convert.ToDouble(double.Parse(totalBytesToReceive.ToString(CultureInfo.Invar
iantCulture))) / 1024 / 1024 / 100));

TaskbarProgress.ProgressValue = Convert.ToDouble(percentTotal) /
100;
progress.Value = percentTotal;
progress_file.Value = percent;

var dSpeed =
(long) (Convert.ToDouble(double.Parse(CurrentBytes2.ToString(CultureInfo.Invar
iantCulture))) / Stopwatch.Elapsed.TotalSeconds);

var received = Utilities.Updater.DetectSize(_currentBytes);
var total = Utilities.Updater.DetectSize(_totalBytes);
var speed = Utilities.Updater.DetectSize(dSpeed);

var downloaded = $"{Загружено ({_count}/{_length}) : {received} /
{total}";

var awaiting = (Convert.ToDouble(_totalBytes - (_currentBytes -
CurrentBytes2)) / 1024) / (Convert.ToDouble(CurrentBytes2) / 1024 /
Stopwatch.Elapsed.TotalSeconds) - (Convert.ToDouble(CurrentBytes2) / 1024) /
(Convert.ToDouble(CurrentBytes2) / 1024 / Stopwatch.Elapsed.TotalSeconds);

labelprogress.Content = $"{downloaded} ({speed}/c)
~{{{(int)(awaiting / 3600):0}} ч {{{(int)(awaiting % 3600 / 60):0}} мин
{{{(awaiting % 3600 % 60):0}} сек";
}

/// <summary>
/// Downloading done dispatcher event
/// </summary>
/// <param name="visible">Show download accesories or not</param>
private void DownloadCompleted(bool visible)
{
    if (visible)
    {
        DownloadBar.Visibility = Visibility.Visible;
        TaskbarProgress.ProgressState =
TaskbarItemProgressState.Normal;
        TaskbarPlay.IsEnabled = false;
        btn_play.IsEnabled = false;
        labelmsg.Content = "Идет обновление...";
    }
    else
    {
        DownloadBar.Visibility = Visibility.Hidden;
        TaskbarProgress.ProgressState =
TaskbarItemProgressState.None;
        TaskbarPlay.IsEnabled = true;
        labelmsg.Content = "Игра обновлена";
    }
}

```

```

        btn_play.IsEnabled = true;
    }
}

/// <summary>
/// Window moving event
/// </summary>
/// <param name="sender"></param>
/// <param name="e"></param>
private void Grid_MouseDown(object sender, MouseButtonEventArgs e)
{
    if (e.LeftButton != MouseButtonState.Pressed) return;
    Cursor = Cursors.SizeAll;
    DragMove();
    Cursor = Cursors.Arrow;
}

/// <summary>
/// Deleting all patches while want leave server
/// </summary>
public void DPatches()
{
    _length = File.ReadAllLines(_tempPath).Length;

    using (var reader = new StreamReader(_tempPath))
    {
        string line;

        while ((line = reader.ReadLine()) != null)
        {
            var ex = line.Split(char.Parse("#"));

            var path = Utilities.Updater.GetPath(_gPath, ex[1]);

            try
            {
                File.Delete(path);
            }
            catch
            {
                // ignored
            }
        }
    }

    File.Delete(_updatePath);

    MessageBox.Show("Все файлы успешно удалены", "Удаление файлов",
        MessageBoxButton.OK, MessageBoxImage.Information);
    btn_play.Visibility = Visibility.Hidden;
    labelmsg.Content = "Запуск клиента невозможен";
}

/// <summary>
/// Set download progress type like just total progress, just current
progress, mixed progres
/// </summary>
/// <param name="index"></param>
public void SetProgressType(int index)
{
    switch (index)
    {
        case 0:
    }
}

```

```

        totalProgressGrid.SetValue(Grid.RowProperty, 1);
        totalProgressGrid.Visibility = Visibility.Visible;
        currentProgressGrid.Visibility = Visibility.Visible;
        break;
    case 1:
        totalProgressGrid.SetValue(Grid.RowProperty, 2);
        totalProgressGrid.Visibility = Visibility.Visible;
        currentProgressGrid.Visibility = Visibility.Hidden;
        break;
    case 2:
        totalProgressGrid.Visibility = Visibility.Hidden;
        currentProgressGrid.Visibility = Visibility.Visible;
        break;
    }
}

private void btn_play_Click(object sender, RoutedEventArgs e)
{
    Play();
}

private void btn_close_Click(object sender, RoutedEventArgs e)
{
    Application.Current.Shutdown();
}

private void Hyperlink_MouseLeftButtonDown(object sender,
MouseEventArgs e)
{
    var hyperlink = (Hyperlink)sender;
    Process.Start(hyperlink.NavigateUri.ToString());
}

private void TaskbarPlay_Click(object sender, EventArgs e)
{
    Play();
}

private void Play()
{
    try
    {
        var process = Process.Start(string.Format(@"{0}\{1}", _gPath,
Properties.Settings.Default.ClientExeName));

        if (Properties.Settings.Default.IsAutoLogin)
        {
            var ni = new NotifyIcon
            {
                Icon = new Icon(
                    Application
                        .GetResourceStream(new
Uri("pack://application:,,,/img/101.ico"))
                        ?.Stream
                    ?? throw new NullReferenceException("Не найден
ресурс 101.ico"))
                ,
                Visible = true
            };

            ni.ShowBalloonTip(2000, "Программа запуска", "Программа
запуска продолжает работать в фоновом режиме. Чтобы развернуть ее,
используйте двойной щелчок левой кнопки мыши", ToolTipIcon.Info);

```

```

Hide();
ni.DoubleClick +=
    delegate
    {
        Show();
        ni.Visible = false;
    };

var accountName = Properties.Settings.Default.Login;
Thread.Sleep(600);

new Thread(() =>
{
    try
    {
        if (process == null) return;

        Thread.CurrentThread.IsBackground = true;

        while (!process.WaitForInputIdle())
        {
        }

        Thread.Sleep(2000);

        foreach (var accNameLetter in accountName)
        {
            SendMessage(process.MainWindowHandle, WmChar,
new IntPtr(accNameLetter), IntPtr.Zero);
            Thread.Sleep(30);
        }

        if
(!string.IsNullOrEmpty(Properties.Settings.Default.Password))
        {
            SendMessage(process.MainWindowHandle,
WmKeyUp, new IntPtr(VkTab), IntPtr.Zero);
            SendMessage(process.MainWindowHandle,
WmKeyDown, new IntPtr(VkTab), IntPtr.Zero);

            foreach (var accPassLetter in
Properties.Settings.Default.Password)
            {
                SendMessage(process.MainWindowHandle,
WmChar, new IntPtr(accPassLetter), IntPtr.Zero);
                Thread.Sleep(30);
            }

            SendMessage(process.MainWindowHandle,
WmKeyUp, new IntPtr(VkReturn), IntPtr.Zero);
            SendMessage(process.MainWindowHandle,
WmKeyDown, new IntPtr(VkReturn), IntPtr.Zero);
        }
        Thread.CurrentThread.Abort();
    }
    catch
    {
        Thread.CurrentThread.Abort();
    }
}).Start();
}
else
    Application.Current.Shutdown();

```

```

    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);

        if (ex.Message.Equals("Не удается найти указанный файл"))
            ShowFolderDialog();
    }
}

/// <summary>
/// Show WindowDialog with blur effects
/// </summary>
/// <param name="modalWindow">Modal window to show</param>
private void ShowModalWithEffect (Window modalWindow)
{
    var blur = new BlurEffect();
    var current = Background;
    blur.Radius = 15;
    Effect = blur;

    if (modalWindow is Settings settings && AnyDownloads)
    {
        settings.BtnDel.IsEnabled = false;
        settings.ResetPath.IsEnabled = false;
        settings.BtnDel.ToolTip = "Не возможно выполнить\пво время
процесса обновления";
    }

    modalWindow.Owner = this;
    modalWindow.ShowDialog();
    Effect = null;
    Background = current;
}

private void Version_MouseDown(object sender, MouseButtonEventArgs e)
{
    // Set motions only on Left button click
    if (e.LeftButton != MouseButtonState.Pressed) return;

    ShowModalWithEffect(new VerUpd());
}

private void news_box_Loaded(object sender, RoutedEventArgs e)
{
}

private void btn_min_Click(object sender, RoutedEventArgs e)
{
    WindowState = WindowState.Minimized;
}

private void btn_settings_Click(object sender, RoutedEventArgs e)
{
    ShowModalWithEffect(new Settings());
    SetProgressType(Properties.Settings.Default.ProgressBarType);
}

private void link_main_Click(object sender, RoutedEventArgs e)
{
    Process.Start("http://horizon-world.net");
}

```

```

private void link_cabinet_Click(object sender, RoutedEventArgs e)
{
    Process.Start("http://horizon-world.net/forum");
}

private void link_registration_Click(object sender, RoutedEventArgs
e)
{
    Process.Start("http://horizon-world.net/reg");
}

private void link_social_Click(object sender, RoutedEventArgs e)
{
    Process.Start("https://vk.com/horizon_wow");
}

private void link_social_Copy_Click(object sender, RoutedEventArgs e)
{
    Process.Start("http://horizon-world.net/minecraft");
}

private void link_social_Copy1_Click(object sender, RoutedEventArgs
e)
{
    Process.Start("https://discord.com/invite/hkMzUPaMuJ");
}

}

}

// NewsLoaderControl
<UserControl x:Class="Launcher.NewsLoaderControl"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
xmlns:mc="http://schemas.openxmlformats.org/markup-
compatibility/2006"
xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
mc:Ignorable="d"
x:Name="NewsControl"
d:DesignWidth="840" d:DesignHeight="480"
MouseLeftButtonDown="NewsControl_MouseLeftButtonDown">

    <UserControl.Resources>
        <Storyboard x:Key="HoverEffectEnter">
            <DoubleAnimationUsingKeyFrames
Storyboard.TargetProperty="(UIElement.Opacity)"
Storyboard.TargetName="LayoutRoot">
                <EasingDoubleKeyFrame KeyTime="0:0:0.4" Value="1"/>
            </DoubleAnimationUsingKeyFrames>
            <DoubleAnimationUsingKeyFrames
Storyboard.TargetProperty="(UIElement.Opacity)"
Storyboard.TargetName="ButtonArrowLeft">
                <EasingDoubleKeyFrame KeyTime="0:0:0.4" Value="1"/>
            </DoubleAnimationUsingKeyFrames>
            <DoubleAnimationUsingKeyFrames
Storyboard.TargetProperty="(UIElement.Opacity)"
Storyboard.TargetName="ButtonArrowRight">
                <EasingDoubleKeyFrame KeyTime="0:0:0.4" Value="1"/>
            </DoubleAnimationUsingKeyFrames>
        </Storyboard>
    </UserControl.Resources>

```

```

        <DoubleAnimationUsingKeyFrames
Storyboard.TargetProperty="(UIElement.Opacity) "
Storyboard.TargetName="BodyBg">
    <EasingDoubleKeyFrame KeyTime="0:0:0.3" Value="1"/>
    </DoubleAnimationUsingKeyFrames>
</Storyboard>
<Storyboard x:Key="HoverEffectLeave">
    <DoubleAnimationUsingKeyFrames
Storyboard.TargetProperty="(UIElement.Opacity) "
Storyboard.TargetName="LayoutRoot">
        <EasingDoubleKeyFrame KeyTime="0:0:0.4" Value="0.9"/>
    </DoubleAnimationUsingKeyFrames>
    <DoubleAnimationUsingKeyFrames
Storyboard.TargetProperty="(UIElement.Opacity) "
Storyboard.TargetName="ButtonArrowLeft">
        <EasingDoubleKeyFrame KeyTime="0:0:0.4" Value="0"/>
    </DoubleAnimationUsingKeyFrames>
    <DoubleAnimationUsingKeyFrames
Storyboard.TargetProperty="(UIElement.Opacity) "
Storyboard.TargetName="ButtonArrowRight">
        <EasingDoubleKeyFrame KeyTime="0:0:0.4" Value="0"/>
    </DoubleAnimationUsingKeyFrames>
    <DoubleAnimationUsingKeyFrames
Storyboard.TargetProperty="(UIElement.Opacity) "
Storyboard.TargetName="BodyBg">
        <EasingDoubleKeyFrame KeyTime="0:0:0.3" Value="0"/>
    </DoubleAnimationUsingKeyFrames>
</Storyboard>

<Storyboard x:Key="ChangeItemsBegin">
    <DoubleAnimationUsingKeyFrames
Storyboard.TargetProperty="(UIElement.Opacity) "
Storyboard.TargetName="LayoutRoot">
        <EasingDoubleKeyFrame KeyTime="0:0:0" Value="1"/>
        <EasingDoubleKeyFrame KeyTime="0:0:0.2" Value="0"/>
    </DoubleAnimationUsingKeyFrames>
</Storyboard>

<Storyboard x:Key="ChangeItemsEnd">
    <DoubleAnimationUsingKeyFrames
Storyboard.TargetProperty="(UIElement.Opacity) "
Storyboard.TargetName="LayoutRoot">
        <EasingDoubleKeyFrame KeyTime="0:0:0" Value="0"/>
        <EasingDoubleKeyFrame KeyTime="0:0:0.2" Value="1"/>
    </DoubleAnimationUsingKeyFrames>
</Storyboard>

<Style x:Key="PlayButton" TargetType="{x:Type Button}"
BasedOn="{x:Null}">
    <Setter Property="Foreground" Value="White"/>
    <Setter Property="Template">
        <Setter.Value>
            <ControlTemplate TargetType="{x:Type Button}">
                <ControlTemplate.Resources>
                    <Storyboard x:Key="MouseHover">
                        <ColorAnimation
Storyboard.TargetProperty="Background.Color" Storyboard.TargetName="Border"
Duration="0:0:0.3" To="#AA36bffe"/>
                    </Storyboard>
                    <Storyboard x:Key="MouseLeave">
                        <ColorAnimation
Storyboard.TargetProperty="Background.Color" Storyboard.TargetName="Border"
Duration="0:0:0.3" To="#66000000"/>
                    </Storyboard>
                </ControlTemplate.Resources>
            </ControlTemplate>
        </Setter.Value>
    </Setter>

```

```

        </Storyboard>
    </ControlTemplate.Resources>
    <Grid x:Name="Grid" Opacity="1">
        <Border x:Name="Border"
BorderThickness="{TemplateBinding BorderThickness}" Padding="{TemplateBinding
Padding}" HorizontalAlignment="Stretch" Width="Auto" CornerRadius="3"
Background="#66000000"/>
        <!--<TextBlock x:Name="ContentText"
Foreground="{TemplateBinding Foreground}" Text="{TemplateBinding Content}"
Margin="{TemplateBinding Padding}" HorizontalAlignment="{TemplateBinding
HorizontalContentAlignment}" VerticalAlignment="{TemplateBinding
VerticalContentAlignment}" Width="Auto"/>-->
        <ContentPresenter x:Name="ContentText" Margin="2"
HorizontalAlignment="Center"
VerticalAlignment="Center"
RecognizesAccessKey="True"/>
    </Grid>

    <ControlTemplate.Triggers>
        <Trigger Property="IsMouseOver" Value="true">
            <!--<Setter Property="Background"
Value="#AA36bffe" TargetName="Border"/>-->
            <Trigger.EnterActions>
                <BeginStoryboard
Storyboard="{StaticResource MouseHover}"/>
            </Trigger.EnterActions>
            <Trigger.ExitActions>
                <BeginStoryboard
Storyboard="{StaticResource MouseLeave}"/>
            </Trigger.ExitActions>
            </Trigger>
            <Trigger Property="IsPressed" Value="true">
                <Setter Value="#7f000000"
Property="Background" TargetName="Border"/>
            </Trigger>
            <Trigger Property="IsEnabled" Value="false">
                <Setter Property="Opacity" Value="0.20"
TargetName="Grid"/>
            </Trigger>
            <Trigger Property="IsEnabled" Value="true"/>
        </ControlTemplate.Triggers>
    </ControlTemplate>
</Setter.Value>
</Setter>
</Style>
</UserControl.Resources>
<UserControl.Triggers>
    <EventTrigger RoutedEvent="UIElement.MouseEnter">
        <BeginStoryboard Storyboard="{StaticResource HoverEffectEnter}"/>
    </EventTrigger>
    <EventTrigger RoutedEvent="UIElement.MouseLeave">
        <BeginStoryboard Storyboard="{StaticResource HoverEffectLeave}"/>
    </EventTrigger>
</UserControl.Triggers>

    <Grid x:Name="LayoutRoot" Opacity="0.9">
        <Image Source="img/news_bg.jpg" Stretch="UniformToFill"/>
        <Label x:Name="NewsIndicatorLabel"
HorizontalContentAlignment="Center" VerticalContentAlignment="Center"
Foreground="{Binding Foreground, ElementName=NewsControl}"
FontFamily="Verdana" FontSize="18">

```

```

        <TextBlock x:Name="NewsIndicatorText" Text="Загрузка новостей..."
TextWrapping="Wrap"/>
    </Label>
    <Grid x:Name="MainGrid" Background="#E5000000" Visibility="Hidden">
        <Image x:Name="NewsImageControl" />
        <Rectangle x:Name="BodyBg" Fill="#99000000" Height="{Binding
ActualHeight, ElementName=StackPanel}" VerticalAlignment="Bottom"
Opacity="0"/>
        <StackPanel x:Name="StackPanel" VerticalAlignment="Bottom">
            <TextBlock x:Name="NewsHeadControl" FontFamily="Vendara"
FontSize="30" Foreground="{Binding Foreground, ElementName=NewsControl}"
TextWrapping="Wrap" Padding="20,0" />
            <TextBlock x:Name="NewsBodyControl" FontFamily="Vendara"
FontSize="14" Foreground="{Binding Foreground, ElementName=NewsControl}"
TextWrapping="Wrap" Padding="10,0"/>
        </StackPanel>
    </Grid>

    <Button x:Name="ButtonArrowLeft" ClipToBounds="True"
HorizontalAlignment="Left" Height="140" Width="40" VerticalAlignment="Center"
FontSize="34.667" FontFamily="Californian FB" Background="#02000000"
Foreground="White" BorderThickness="0" IsTabStop="False"
IsHitTestVisible="True" Focusable="False" Style="{DynamicResource
PlayButton}" Margin="-5" Click="BtnLeft_Click">
        <Path Width="20" Height="25" Canvas.Left="22.1667"
Canvas.Top="22.1667" Stretch="Fill" Fill="{Binding Foreground,
ElementName=ButtonArrowLeft}" Data="F1 M 33.6458,38L 49.4792,53.8333L
38.7917,53.8333L 22.1667,38L 38.7917,22.1667L 49.4792,22.1667L 33.6458,38 Z
"/>
    </Button>
    <Button x:Name="ButtonArrowRight" ClipToBounds="True"
HorizontalAlignment="Right" Height="140" Width="40"
VerticalAlignment="Center" FontSize="25" FontFamily="SimHei"
Background="#02000000" Foreground="White" BorderThickness="0"
IsTabStop="False" IsHitTestVisible="True" Focusable="False"
Style="{DynamicResource PlayButton}" Margin="-5" Click="BtnRight_Click">
        <Path Width="20" Height="25" Canvas.Left="26.5208"
Canvas.Top="22.1667" Stretch="Fill" Fill="{Binding Foreground,
ElementName=ButtonArrowRight}" Data="F1 M 42.3542,38L 26.5208,53.8333L
37.2083,53.8333L 53.8333,38L 37.2083,22.1667L 26.5208,22.1667L 42.3542,38 Z
"/>
    </Button>
</Grid>
</UserControl>

// NewsLoaderControl.xaml.cs

using System;
using System.Collections.Generic;
using System.Diagnostics;
using System.IO;
using System.Net;
using System.Windows.Threading;
using System.Windows;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Animation;
using System.Windows.Media.Imaging;

namespace Launcher
{
    /// <summary>
    /// Логика взаимодействия для NewsLoaderControl.xaml

```

```

/// </summary>
public partial class NewsLoaderControl
{
    private readonly string _tempPath = Path.GetTempFileName();
    private readonly LinkedList<News> _newsList = new LinkedList<News>();
    private readonly DispatcherTimer _itemChanger = new
DispatcherTimer();
    private LinkedListNode<News> _node;

    private class News
    {
        public ImageSource Image { get; }
        public string Head { get; }
        public string Body { get; }
        public string Link { get; }

        public News(string head, string body, ImageSource image, string
link)
        {
            Head = head;
            Body = body;
            Image = image;
            Link = link;
        }
    }

    public NewsLoaderControl()
    {
        InitializeComponent();
    }

    public ImageSource NewsImage
    {
        get => NewsImageControl.Source;
        set => NewsImageControl.Source = value;
    }

    public string NewsHead
    {
        get => NewsHeadControl.Text;
        set => NewsHeadControl.Text = value;
    }

    public string NewsBody
    {
        get => NewsBodyControl.Text;
        set => NewsBodyControl.Text = value;
    }

    public string NewsLink { get; set; }

    private void NewsControl_MouseLeftButtonDown(object sender,
MouseButtonEventArgs e)
    {
        if (_newsList.Count != 0)
            Process.Start(NewsLink);
    }

    /// <summary>
    /// Смена отображаемой новости
    /// </summary>
    /// <param name="next">Если значение true то устанавливает следующим
элементом LinkedListNode.Next</param>

```

```

private void ChangeNewsItem(bool next)
{
    if (next)
    {
        _node = _node.Next ?? _newsList.First;
    }
    else
    {
        _node = _node.Previous ?? _newsList.Last;
    }

    var sb = FindResource("ChangeItemsBegin") as Storyboard;
    Storyboard.SetTarget(sb ?? throw new
InvalidOperationException("Не найден ресурс \"ChangeItemsBegin\""),
MainGrid);
    sb.Completed += StoryBoard_Completed;
    sb.Begin();

}

private void StoryBoard_Completed(object sender, EventArgs e)
{
    var sb = FindResource("ChangeItemsEnd") as Storyboard;
    Storyboard.SetTarget(sb ?? throw new
InvalidOperationException("Не найден ресурс \"ChangeItemsEnd\""), MainGrid);
    sb.Begin();

    SetNewsValues();
}

public void SetNews(string url)
{
    var client = new WebClient();

    client.DownloadFileCompleted += Client_DownloadFileCompleted;

    client.DownloadFileAsync(new Uri(url), _tempPath);
}

private void Client_DownloadFileCompleted(object sender,
System.ComponentModel.AsyncCompletedEventArgs e)
{
    using (var reader = new StreamReader(_tempPath))
    {
        string line;

        while ((line = reader.ReadLine()) != null)
        {
            var ex = line.Split(char.Parse("#"));
            var imageUri = new Uri(ex[2]);
            var imageBitmap = new BitmapImage(imageUri);

            var news = new News(ex[0], ex[1], imageBitmap, ex[3]);

            _newsList.AddLast(news);
        }
    }

    if (_newsList.Count != 0)
    {
        MainGrid.Visibility = Visibility.Visible;
        NewsIndacatorLabel.Visibility = Visibility.Hidden;
    }
}

```

```

        _node = _newsList.First;
        SetNewsValues();

        _itemChanger.Tick += ItemChanger_Tick;
        _itemChanger.Interval = new TimeSpan(0, 0, 5);
        _itemChanger.Start();
    }
    else
    {
        NewsIndicatorText.Text = "Список новостей пуст";
        ButtonArrowLeft.Visibility = Visibility.Hidden;
        ButtonArrowRight.Visibility = Visibility.Hidden;
    }
}

private void ItemChanger_Tick(object sender, EventArgs e)
{
    ChangeNewsItem(true);
}

private void SetNewsValues()
{
    NewsImage = _node.Value.Image;
    NewsHead = _node.Value.Head;
    NewsBody = _node.Value.Body;
    NewsLink = _node.Value.Link;
}

private void BtnRight_Click(object sender, RoutedEventArgs e)
{
    ChangeNewsItem(true);
    _itemChanger.Interval = new TimeSpan(0, 0, 5);
    _itemChanger.Stop();
    _itemChanger.Start();
}

private void BtnLeft_Click(object sender, RoutedEventArgs e)
{
    ChangeNewsItem(false);
    _itemChanger.Interval = new TimeSpan(0, 0, 5);
    _itemChanger.Stop();
    _itemChanger.Start();
}
}

// Settings
<Window
xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
mc:Ignorable="d" x:Name="Window" x:Class="Launcher.Settings"
Title="Settings" Height="485" Width="540" AllowsTransparency="True"
ResizeMode="NoResize" ShowInTaskbar="False"
WindowStartupLocation="CenterScreen" Topmost="True" WindowStyle="None"
Background="#00000000" Foreground="White" Loaded="Window_Loaded">
    <Grid>
        <Rectangle Stroke="Black" RadiusY="3" RadiusX="3">
            <Rectangle.Fill>
                <LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">
                    <GradientStop Color="#FF6F7277" Offset="0.003" />

```

```

        <GradientStop Color="#FF3F4144" Offset="0.003" />
        <GradientStop Color="#FF010204" Offset="0.3" />
        <!--<GradientStop Color="#FF010204" Offset="0.7" />
        <GradientStop Color="#FF3F4144" Offset="0.995" />
        <GradientStop Color="#FF6F7277" Offset="0.995"
/>-->

        </LinearGradientBrush>
        </Rectangle.Fill>
    </Rectangle>
    <ScrollViewer HorizontalAlignment="Left" Height="363"
VerticalAlignment="Top" Width="520" Margin="10,55,0,0" Background="#FF191919"
BorderBrush="#FF191919">
        <StackPanel >
            <Grid Height="111" Width="480" Margin="10">

                <Rectangle Stroke="Black" RadiusY="3" RadiusX="3">
                    <Rectangle.Fill>
                        <LinearGradientBrush EndPoint="0.5,1"
StartPoint="0.5,0">
                            <GradientStop Color="#FF6F7277"
Offset="0.995" />
                            <GradientStop Color="#FF010204" />
                            <GradientStop Color="#FF3F4144"
Offset="0.994" />
                        </LinearGradientBrush>
                    </Rectangle.Fill>
                </Rectangle>
                <Label Content="Настройка клиента игры"
HorizontalAlignment="Left" Width="386" Foreground="White" FontWeight="Bold"
FontSize="15" Height="32" VerticalAlignment="Top" />
                <Label Content="Путь" HorizontalAlignment="Left"
Margin="10,37,0,0" VerticalAlignment="Top" Width="49" Foreground="{Binding
Foreground, ElementName=Window}" FontSize="15"/>
                <TextBox x:Name="Folder" Height="30" Margin="64,37,52,0"
VerticalAlignment="Top" Background="#19000000" BorderBrush="#B2171717"
SelectionBrush="#FF5F5F5F" Foreground="#FFE2E2E2" FontSize="15"
VerticalContentAlignment="Center" AllowDrop="True" IsTabStop="False"
BorderThickness="0" IsReadOnly="True"/>
                <Button Content="..." HorizontalAlignment="Left"
Margin="433,37,0,0" VerticalAlignment="Top" Width="37" Height="30"
Style="{DynamicResource PlayButton}" Click="Button_Click_1"
IsTabStop="False"/>
                <Button Content="Сброс" x:Name="ResetPath"
HorizontalAlignment="Left" Margin="395,72,0,0" VerticalAlignment="Top"
Width="75" Height="30" Style="{DynamicResource PlayButton}" IsTabStop="False"
Click="ResetPath_Click" ToolTipService.ShowOnDisabled="True"/>
            </Grid>
            <Grid Height="186" Width="480" Margin="10">

                <Rectangle Stroke="Black" RadiusY="3" RadiusX="3">
                    <Rectangle.Fill>
                        <LinearGradientBrush EndPoint="0.5,1"
StartPoint="0.5,0">
                            <GradientStop Color="#FF6F7277"
Offset="0.995" />
                            <GradientStop Color="#FF010204" />
                            <GradientStop Color="#FF3F4144"
Offset="0.994" />
                        </LinearGradientBrush>
                    </Rectangle.Fill>
                </Rectangle>

```

```

        <Label Content="Настройка входа"
HorizontalAlignment="Left" Width="386" Foreground="White" FontWeight="Bold"
FontSize="15" Height="32" VerticalAlignment="Top" />
        <CheckBox x:Name="Autologin" Content="Автоматическая
авторизация в клиенте" HorizontalAlignment="Left" Margin="10,63,0,0"
VerticalAlignment="Top" Foreground="White" FontSize="15" Height="17"
Width="307" IsTabStop="False"/>
        <Label Content="Логин пользователя"
HorizontalAlignment="Left" Margin="10,101,0,0" VerticalAlignment="Top"
Width="165" Foreground="{Binding Foreground, ElementName=Window}"
FontSize="15"/>
        <TextBox x:Name="Login" Height="30" Margin="180,101,10,0"
TextWrapping="Wrap" VerticalAlignment="Top" Background="#19000000"
BorderBrush="#B2171717" SelectionBrush="#FF5F5F5F" Foreground="#FFE2E2E2"
FontSize="15" VerticalContentAlignment="Center"
HorizontalContentAlignment="Center" IsTabStop="False" BorderThickness="0"/>
        <Label Content="Пароль" HorizontalAlignment="Left"
Margin="10,145,0,0" VerticalAlignment="Top" Width="165" Foreground="{Binding
Foreground, ElementName=Window}" FontSize="15"/>
        <PasswordBox x:Name="Password" Height="30"
Margin="180,145,10,0" VerticalAlignment="Top" Background="#19000000"
BorderBrush="#B2171717" SelectionBrush="#FF5F5F5F" Foreground="#FFE2E2E2"
FontSize="15" VerticalContentAlignment="Center"
HorizontalContentAlignment="Center" IsTabStop="False" BorderThickness="0"/>

    </Grid>
    <Grid Height="133" VerticalAlignment="Top" Width="480"
Margin="10">
        <Rectangle Stroke="Black" RadiusY="3" RadiusX="3">
            <Rectangle.Fill>
                <LinearGradientBrush EndPoint="0.5,1"
StartPoint="0.5,0">
                    <GradientStop Color="#FF6F7277"
Offset="0.995" />
                    <GradientStop Color="#FF010204" />
                    <GradientStop Color="#FF3F4144"
Offset="0.994" />
                </LinearGradientBrush>
            </Rectangle.Fill>
        </Rectangle>
        <Label Content="Уходя с сервера"
HorizontalAlignment="Left" Margin="2,0,0,0" Width="386" Foreground="White"
FontWeight="Bold" FontSize="15" Height="32" VerticalAlignment="Top" />
        <Button x:Name="BtnDel" Cursor="{Binding Cursor,
ElementName=version}" Visibility="Visible" Content="Удалить файлы сервера"
HorizontalAlignment="Left" Margin="252,0,0,10" Width="218"
Style="{DynamicResource PlayButton}" FontSize="16" IsTabStop="False"
BorderThickness="1" FontFamily="Verdana" Height="45"
VerticalAlignment="Bottom" ToolTipService.ShowOnDisabled="True"
Click="ButtonDelete_Click"/>
        <Label HorizontalAlignment="Left" Height="74"
Margin="10,49,0,0" VerticalAlignment="Top" Width="237" Foreground="White">
            <TextBlock><Run FontWeight="Light" Text="Если вы
захотите уйти с сервера," /><LineBreak/><Run FontWeight="Light" Text="то вы
просто можете нажать на эту кнопку," /><LineBreak/><Run FontWeight="Light"
Text="что позволит удалить все скаченные файлы" /><LineBreak/><Run
FontWeight="Light" Text="не повредив клиент игры." /></TextBlock>
        </Label>

    </Grid>
    <Grid Height="147" VerticalAlignment="Top" Width="480">
        <Rectangle Stroke="Black" RadiusY="3" RadiusX="3">
            <Rectangle.Fill>

```

```

        <LinearGradientBrush EndPoint="0.5,1"
StartPoint="0.5,0">
            <GradientStop Color="#FF6F7277"
Offset="0.995" />
            <GradientStop Color="#FF010204" />
            <GradientStop Color="#FF3F4144"
Offset="0.994" />
        </LinearGradientBrush>
    </Rectangle.Fill>
</Rectangle>
    <Label Content="Настройка загрузки"
HorizontalAlignment="Left" Margin="2,0,0,0" Width="386" Foreground="White"
FontWeight="Bold" FontSize="15" Height="32" VerticalAlignment="Top" />
    <Label HorizontalAlignment="Left" Height="35"
Margin="10,38,0,0" VerticalAlignment="Top" Width="301" Foreground="{Binding
Foreground, ElementName=Window}" Content="Выберите вариант отображения
загрузки:" FontSize="15"/>
    <ComboBox x:Name="ProgressType" HorizontalAlignment="Left"
Margin="311,44,0,0" VerticalAlignment="Top" Width="148" IsTabStop="False"
Height="29" VerticalContentAlignment="Center">
        <ComboBoxItem Content="Смешанная"/>
        <ComboBoxItem Content="Общая"/>
        <ComboBoxItem Content="Пофайловая"/>
    </ComboBox>
    <Label HorizontalAlignment="Left" Height="30"
Margin="10,78,0,0" VerticalAlignment="Top" Width="301" Foreground="{Binding
Foreground, ElementName=Window}" Content="Выберите значение ограничения
загрузки:" FontSize="14.667"/>

    <TextBox x:Name="SpeedValue" Height="30"
Margin="311,78,92,0" VerticalAlignment="Top" Background="#19000000"
BorderBrush="#B2171717" SelectionBrush="#FF5F5F5F" Foreground="#FFE2E2E2"
FontSize="15" VerticalContentAlignment="Center" AllowDrop="True"
IsTabStop="False" BorderThickness="0"
PreviewTextInput="SpeedValue_PreviewTextInput"
HorizontalContentAlignment="Right"/>
    <ComboBox x:Name="SpeedType" HorizontalAlignment="Left"
Margin="393,78,0,0" VerticalAlignment="Top" Width="66" IsTabStop="False"
Height="29" VerticalContentAlignment="Center">
        <ComboBoxItem Content="КБ/с"/>
        <ComboBoxItem Content="МБ/с"/>
        <ComboBoxItem Content="ГБ/с"/>
    </ComboBox>
    <Label Content="0 - ограничение отсутствует"
HorizontalAlignment="Left" Height="24" Margin="10,113,0,0"
VerticalAlignment="Top" Width="449" Foreground="#FF838383"/>
</Grid>
</StackPanel>
</ScrollViewer>
<Button x:Name="ButtonSave" Content="OK" HorizontalAlignment="Right"
Height="42" Margin="0,0,10,10" VerticalAlignment="Bottom" Width="98"
FontSize="15" Style="{DynamicResource PlayButton}" Click="ButtonSave_Click"
IsTabStop="False"/>
    <Label Content="Настройка программы" Margin="46,10,69,0"
Foreground="White" FontWeight="Bold" FontSize="20" Height="40"
VerticalAlignment="Top" >
    <Label.Effect>
        <DropShadowEffect BlurRadius="0" ShadowDepth="1"
Direction="135"/>
    </Label.Effect>
</Label>

```

```

    <Label Margin="1,0,0,0" Foreground="White" FontWeight="Bold"
    FontSize="15" VerticalAlignment="Top" HorizontalAlignment="Left" Height="50"
    Width="50" >
        <Path
            Width="40"
            Height="40"
            SnapsToDevicePixels="False"
            StrokeThickness="0"
            Stretch="Fill"
            Fill="White"
            Data="F1 M 32.6227,68.9728C 33.1208,70.2446
33.7077,71.4719 34.3757,72.6471C 35.567,72.7702 36.7761,72.8333 38,72.8333C
40.6447,72.8333 43.2205,72.5386 45.6965,71.9802C 42.2807,68.5449
40.1698,63.8107 40.1698,58.5835C 40.1698,48.0901 48.6764,39.5835
59.1698,39.5835C 64.2047,39.5835 68.7822,41.5419 72.182,44.7385C
72.6094,42.5584 72.8333,40.3053 72.8333,38C 72.8333,36.442 72.731,34.908
72.5328,33.4041C 70.4772,32.3109 68.2718,31.4621 65.9554,30.8963L
65.6219,22.7367C 63.5664,22.3692 61.4511,22.1741 59.2916,22.1671L
57.491,30.1322C 54.2845,30.3184 51.2204,31.0348 48.3848,32.1949L
43.318,25.7887C 41.3995,26.7177 39.5749,27.8102 37.8622,29.0479L
41.0883,36.5527C 38.6742,38.5364 36.5919,40.9091 34.9373,43.575L
27.0703,41.3697C 26.073,43.2256 25.2319,45.178 24.5646,47.2093L
31.5867,51.385C 30.9883,53.6844 30.6698,56.0968 30.6698,58.5835L
30.746,60.6839L 23.0854,63.5239C 23.3743,65.6539 23.8474,67.7255
24.4881,69.7219L 32.6227,68.9728 Z M 27.0669,37.1755L 34.9338,39.3807C
35.9312,37.5248 36.7723,35.5724 37.4395,33.5412L 30.4175,29.3654C
31.0159,27.066 31.3344,24.6537 31.3344,22.1669L 31.2581,20.0665L
38.9187,17.2265C 38.6299,15.0965 38.1568,13.0249 37.5161,11.0285L
29.3815,11.7776C 28.4787,9.47258 27.2844,7.31377 25.8439,5.34655C
22.7943,6.48232 19.9467,8.03307 17.3698,9.93015C 20.1557,13.2359
21.8344,17.5054 21.8344,22.1669C 21.8344,32.5019 13.5826,40.9096
3.30816,41.1611C 3.60321,44.4411 4.35293,47.59 5.49381,50.5444C
8.34489,50.2806 11.0742,49.5968 13.6194,48.5555L 18.6861,54.9617C
20.6046,54.0327 22.4293,52.9403 24.142,51.7025L 20.9158,44.1977C
23.3299,42.2141 25.4123,39.8414 27.0669,37.1755 Z "
            <Path.Effect>
                <DropShadowEffect ShadowDepth="1" BlurRadius="0"
                Direction="135"/>
            </Path.Effect>
        </Path>
    </Label>
    <Rectangle Height="20" Margin="10,55,10,0" VerticalAlignment="Top"
    IsHitTestVisible="False">
        <Rectangle.Fill>
            <LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">
                <GradientStop Color="#FF191919" Offset="0"/>
                <GradientStop Color="#00191919" Offset="1"/>
            </LinearGradientBrush>
        </Rectangle.Fill>
    </Rectangle>
    <Rectangle Height="20" Margin="10,0,10,67" VerticalAlignment="Bottom"
    IsHitTestVisible="False">
        <Rectangle.Fill>
            <LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">
                <GradientStop Color="#FF191919" Offset="1"/>
                <GradientStop Color="#00191919" Offset="0"/>
            </LinearGradientBrush>
        </Rectangle.Fill>
    </Rectangle>

</Grid>
</Window>

```

```

//Settings.xaml.cs
using System;
using System.IO;
using System.Windows;
using System.Text.RegularExpressions;
using Launcher.HelpClasses;

namespace Launcher
{
    public partial class Settings
    {
        public Settings()
        {
            InitializeComponent();
        }

        private void ButtonSave_Click(object sender, RoutedEventArgs e)
        {
            Properties.Settings.Default.IsAutoLogin = (Autologin.IsChecked ==
true);
            Properties.Settings.Default.Login = Login.Text;
            Properties.Settings.Default.Password = Password.Password;
            Properties.Settings.Default.ProgressBarType =
ProgressType.SelectedIndex;
            Properties.Settings.Default.GameFolder = Folder.Text;
            Properties.Settings.Default.DownloadSpeedLimit =
(long) (Convert.ToDouble(SpeedValue.Text) * Math.Pow(1024,
SpeedType.SelectedIndex + 1));
            Properties.Settings.Default.Save();

            if (Owner is MainWindow m && m.AnyDownloads)
            {
                m.CurrentBytes2 = 0;
                m.CurrentFileBytes2 = 0;

                m.StopWatch.Stop();
                m.StopWatch.Reset();
                m.StopWatch.Start();
            }
            Close();
        }

        private void Window_Loaded(object sender, RoutedEventArgs e)
        {
            Login.Text = Properties.Settings.Default.Login;
            Password.Password = Properties.Settings.Default.Password;
            Autologin.IsChecked = Properties.Settings.Default.IsAutoLogin;
            ProgressType.SelectedIndex =
Properties.Settings.Default.ProgressBarType;
            Folder.Text = Properties.Settings.Default.GameFolder;

            SpeedValue.Text =
Utilities.Updater.DetectSize(Properties.Settings.Default.DownloadSpeedLimit,
out var index);
            SpeedType.SelectedIndex = index;
        }

        private void Button_Click_1(object sender, RoutedEventArgs e)
        {

```

```

var folder = new System.Windows.Forms.FolderBrowserDialog
{
    Description = @"Выберите папку с клиентом игры",
    RootFolder = Environment.SpecialFolder.MyComputer,
    ShowNewFolderButton = false
};
var result = folder.ShowDialog();
if (result != System.Windows.Forms.DialogResult.OK) return;

if (File.Exists(Path.Combine(folder.SelectedPath, "Wow.exe")))
{
    var folderPath = folder.SelectedPath;
    Folder.Text = folderPath;
}
else
{
    MessageBox.Show("Файл \"Wow.exe\" не найден!\nПожалуйста
выберите корректную папку с игрой!", "Ошибка выбора папки",
MessageBoxButton.OK, MessageBoxImage.Error);
}

private void ResetPath_Click(object sender, RoutedEventArgs e)
{
    Folder.Text = "Не задано";
}

private void ButtonDelete_Click(object sender, RoutedEventArgs e)
{
    if (MessageBox.Show("Вы действительно хотите удалить все
загруженные файлы сервера?", "Подтверждение удаления",
MessageBoxButton.YesNoCancel, MessageBoxImage.Question) ==
MessageBoxResult.Yes)
    {
        if (Owner is MainWindow m) m.DPatches();
    }
}

private void SpeedValue_PreviewTextInput(object sender,
System.Windows.Input.TextCompositionEventArgs e)
{
    e.Handled = !IsTextAllowed(e.Text);
}

private static bool IsTextAllowed(string text)
{
    var regex = new Regex("[^0-9]+");
    return !regex.IsMatch(text);
}
}
// VerUpd
<Window x:Class="Launcher.VerUpd"
xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
Title="Описание обновлений инструментария" AllowsTransparency="True"
Height="300" Width="480" ResizeMode="NoResize" ShowInTaskbar="False"
WindowStartupLocation="CenterScreen" Topmost="True" WindowStyle="None"
Background="#00000000">
    <Grid>
        <Rectangle Stroke="Black" RadiusY="3" RadiusX="3">
            <Rectangle.Fill>
                <LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

```

```

        <GradientStop Color="#FF6F7277" Offset="0.005" />
        <GradientStop Color="#FF010204" Offset="0.3" />
        <GradientStop Color="#FF3F4144" Offset="0.005" />
    </LinearGradientBrush>
</Rectangle.Fill>
</Rectangle>
    <Button x:Name="BtnClose" Margin="0,8,10,0" Style="{DynamicResource
PlayButton}" Click="BtnClose_Click" FontSize="16" Height="28"
VerticalAlignment="Top" HorizontalAlignment="Right" Width="28"
ToolTip="Закрѳть окно" Padding="1" >
    <Path
        Width="16"
        Height="16"
        x:Name="CheckMark"
        SnapsToDevicePixels="False"
        StrokeThickness="2"
        Stroke="White"
        Data="M 0 0 L 7 7 M 0 7 L 7 0" Stretch="Fill">
    </Path>
</Button>
    <RichTextBox x:Name="NewsBox" Margin="12,41,12,12"
AcceptsReturn="False" IsReadOnly="True" ScrollViewer.CanContentScroll="True"
VerticalScrollBarVisibility="Auto" Background="#A5000000" BorderBrush="Black"
Foreground="#FFE8E8E8" Focusable="False" IsTabStop="False"
Template="{DynamicResource NewsBox}" Cursor="Arrow">
    <RichTextBox.Effect>
        <DropShadowEffect Color="#FFA2A2A2" Direction="0"
ShadowDepth="0" Opacity="0.5"/>
    </RichTextBox.Effect>
    <RichTextBox.Resources>
        <Style TargetType="Hyperlink">
            <Setter Property="Cursor" Value="Hand" />
            <EventSetter Event="MouseLeftButtonDown"
Handler="Hyperlink_MouseLeftButtonDown" />
        </Style>
    </RichTextBox.Resources>
    <FlowDocument>
        </FlowDocument>
    </RichTextBox>
    <Label Content="Описание обновлений программы"
HorizontalAlignment="Left" Margin="12,8,0,0" Width="420" Foreground="White"
FontWeight="Bold" FontSize="15" Height="32" VerticalAlignment="Top"
IsHitTestVisible="False" />
</Grid>
</Window>

// VerUpd.xaml.cs
<Window x:Class="Launcher.VerUpd"
xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
Title="Описание обновлений инструментария" AllowsTransparency="True"
Height="300" Width="480" ResizeMode="NoResize" ShowInTaskbar="False"
WindowStartupLocation="CenterScreen" Topmost="True" WindowStyle="None"
Background="#00000000">
    <Grid>
        <Rectangle Stroke="Black" RadiusY="3" RadiusX="3">
            <Rectangle.Fill>
                <LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">
                    <GradientStop Color="#FF6F7277" Offset="0.005" />
                    <GradientStop Color="#FF010204" Offset="0.3" />
                    <GradientStop Color="#FF3F4144" Offset="0.005" />
                </LinearGradientBrush>
            </Rectangle.Fill>
        </Rectangle>
    </Grid>

```

```

        </LinearGradientBrush>
        </Rectangle.Fill>
    </Rectangle>
    <Button x:Name="BtnClose" Margin="0,8,10,0" Style="{DynamicResource
PlayButton}" Click="BtnClose_Click" FontSize="16" Height="28"
VerticalAlignment="Top" HorizontalAlignment="Right" Width="28"
ToolTip="Заккрыть окно" Padding="1" >
        <Path
            Width="16"
            Height="16"
            x:Name="CheckMark"
            SnapsToDevicePixels="False"
            StrokeThickness="2"
            Stroke="White"
            Data="M 0 0 L 7 7 M 0 7 L 7 0" Stretch="Fill">
        </Path>
    </Button>
    <RichTextBox x:Name="NewsBox" Margin="12,41,12,12"
AcceptsReturn="False" IsReadOnly="True" ScrollViewer.CanContentScroll="True"
VerticalScrollBarVisibility="Auto" Background="#A5000000" BorderBrush="Black"
Foreground="#FFE8E8E8" Focusable="False" IsTabStop="False"
Template="{DynamicResource NewsBox}" Cursor="Arrow">
        <RichTextBox.Effect>
            <DropShadowEffect Color="#FFA2A2A2" Direction="0"
ShadowDepth="0" Opacity="0.5"/>
        </RichTextBox.Effect>
        <RichTextBox.Resources>
            <Style TargetType="Hyperlink">
                <Setter Property="Cursor" Value="Hand" />
                <EventSetter Event="MouseLeftButtonDown"
Handler="Hyperlink_MouseLeftButtonDown" />
            </Style>
        </RichTextBox.Resources>
        <FlowDocument>
            </FlowDocument>
        </RichTextBox>
        <Label Content="Описание обновлений программы"
HorizontalAlignment="Left" Margin="12,8,0,0" Width="420" Foreground="White"
FontWeight="Bold" FontSize="15" Height="32" VerticalAlignment="Top"
IsHitTestVisible="False" />
    </Grid>
</Window>

```