

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество  
АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ

кафедра компьютерных технологий

«Допущен к защите»  
Заведующий кафедрой \_\_\_\_\_  
(Ф.И.О., ученая степень, звание)  
«    » 20\_\_ г.  
(подпись)

ДИПЛОМНЫЙ ПРОЕКТ

На тему: Мобильное приложение

Специальность Вычислительная техника и информатическое обеспечение

Выполнил (а) Шубин М.А. BT-12-2  
(Фамилия и инициалы) группа

Научный руководитель Рахымбергенов С.Р. доцент, к.т.н.  
(Фамилия и инициалы, ученая степень, звание)

Консультанты:

по экономической части:

Бектешева А.У., к.э.н., доцент  
(Фамилия и инициалы, ученая степень, звание)  
АУ « 04 » 06 2016 г.  
(подпись)

по безопасности жизнедеятельности:

Приходко Н.И. д.т.н., профессор  
(Фамилия и инициалы, ученая степень, звание)  
НП « 02 » 06 2016 г.  
(подпись)

по применению вычислительной техники:

Рахымбергенов С.Р. доцент к.т.н.  
(Фамилия и инициалы, ученая степень, звание)  
СР « 31 » мая 2016 г.  
(подпись)

Нормоконтролер: Мусатрова Т.Д. к.т.н. ст. преподаватель  
(Фамилия и инициалы, ученая степень, звание)  
«    » 20\_\_ г.  
(подпись)

Рецензент: Согдатов Б.Д. док. наук  
(Фамилия и инициалы, ученая степень, звание)  
«    » 20\_\_ г.  
(подпись)

Алматы 2016 г.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество  
АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ

Факультет Аэрокосмических и информационных технологий  
Специальность 58070400 - Вычислительная техника и информатизация обслуживания  
Кафедра компьютерных технологий

ЗАДАНИЕ

на выполнение дипломного проекта

Студент Шубин Михаил Александрович  
(фамилия, имя, отчество)

Тема проекта Мобильное игровое приложение

утверждена приказом ректора № 148 от «19» октября 2015 г.  
Срок сдачи законченной работы «\_\_» \_\_\_\_\_ 20\_\_ г.  
Исходные данные к проекту требуемые параметры результатов проектирования (исследования) и исходные данные объекта

Требуется разработка игрового мобильного приложения.  
Плюс реализовать игровую логику. Также изучить  
построение игровой механики.

Перечень подлежащих разработке дипломного проекта вопросов или краткое содержание дипломного проекта:

1. Определить подходящий язык программирования с  
предпочтением к бесплатному
2. Спланировать архитектуру игры на различные  
платформы
3. Разработать звуковую графику и создание звукового  
сопровождения
4. Разработать мобиль и расширенную архитектуру  
игрового приложения.

Перечень графического материала (с точным указанием обязательных чертежей)

В расчетной части приведены диаграммы построения игрового движка




В экономической части приведены диаграммы роста и тенденции игрового рынка и экономические обоснования.

В разделе о безопасности и жизнедеятельности приведены схемы помещений, которые кондиционируются и освещаются.

Рекомендуемая основная литература

1. Scott Rogers, Level Up The Guide to great video game design
2. Шаблоны проектирования, Ханна
3. Jesse Schell, The art of game design
4. Robert Nyström, game programming patterns

Консультанты по проекту с указанием относящихся к ним разделов


Раздел	Консультант	Сроки	Подпись
Вычислительная техника	Радимовичев С.Р.		
Безопасность жизнедеятельности	Триходько Р.П.	18.03 - 02.06.16	
Экономическая часть	Бекмиева А.И.	18.03 - 02.06.16	


**Г Р А Ф И К**  
подготовки дипломного проекта

№ п/п	Наименование разделов, перечень разрабатываемых вопросов	Сроки представления руководителю	Примечание
1.	Демонстрация вычислительной части проекта трехмерной библиотеки	03.02.16	
2.	Предварительное моделирование игрового движка	11.03.16	
3.	Демонстрация базовой игровой механики	24.03.16	
4.	Работы по механике	16.04.16	
5.	Готовый продукт со всеми пояснениями и рисунками	3.05.16	

Дата выдачи задания « 24 » сентября 2016 г.

Заведующий кафедрой \_\_\_\_\_  
(подпись) (Фамилия и инициалы)

Руководитель  \_\_\_\_\_  
(подпись) Раменбергская С.В.  
(Фамилия и инициалы)

Задание принял к исполнению студент  \_\_\_\_\_  
(подпись) Шубин М.А.  
(Фамилия и инициалы)

## **Аннотация**

В данной работе была разработана игра. Была выполнена работа по портированию проекта на разные платформы. Проведен анализ на подбор интересной игровой механики. Произведен анализ игровой индустрии в целом и экономической эффективности продукта, также был оценен местный рынок. Вся работа выполнена в соответствии с техническими требованиями и требованиями безопасности.

## **Abstract**

The game was developed in this paper. Work has been done on porting teachers to cooperate on different platforms. The analysis for the selection of an interesting game mechanics. The analysis of the gaming industry in general and the economic efficiency of the product, and the local market has been estimated. All work is performed in accordance with the technical and safety requirements.

## **Аңдатпа**

Бұл жұмыса ойын құралған. Жобаны түрлі платформаларға порттау жұмысы жасалған. Қызықты ойын таңдау механикасы жүргізілген. Жалпы ойын өнеркәсіп талдау және өнімнің экономикалық тиімділігін, сондай-ақ жергілікті нарық бағаланған. Барлық жұмыстар техникалық және қауіпсіздік талаптарына сәйкес жүзеге асырылады.

## Содержание

Введение .....	12
1. Особенности выбора технологий .....	14
1.1 Язык программирования С#.....	14
1.1.1 Роль платформы .NET .....	15
1.1.2 Архитектура платформы .NET Framework.....	16
1.1.3 Управляемый и неуправляемый код .....	17
1.1.4 JIT-компиляция .....	18
1.2 Платформа Mono/Хamarin?.....	18
1.3 Проект XNA/Monogame?.....	20
2 Компьютерные игры .....	24
2.1 Игровые платформы .....	24
2.2 Игровые движки.....	24
2.3 Разработка игр .....	25
2.4 Издание и оперирование игр.....	25
2.5 Популяризация (маркетинг игр).....	25
2.6 Игровые сообщества.....	26
2.7 Киберспорт .....	26
2.8 Творчество игроков.....	26
2.9 Индустрия компьютерных игр: история зарождения и развития .....	26
2.10 Современность и прогнозы .....	28
2.11 Игры как искусство.....	30
3. Разработка компьютерных игр.....	34
3.1 Предпроизводственный процесс .....	35
3.2 Производство .....	36
3.3 Поддержка.....	37
3.4 Игровые профессии.....	38
3.4.1 Программисты .....	38
3.4.2 Художники .....	38
3.4.3 Музыканты.....	38
3.4.4 Писатели.....	39
3.4.5 Дизайнеры уровней .....	39
3.4.6 Тестеры.....	39
3.5 Перспективы программиста компьютерных игр .....	39
4 Экономическое обоснование проекта .....	40
4.1 Описание работы и обоснование необходимости .....	40
4.2 Трудовые ресурсы, используемые в работе. ....	44
4.3 Оборудование, используемое в работе .....	44

4.4 Программное обеспечение, используемое в работе .....	45
4.5 Сроки реализации проекта .....	46
4.6 Расчет стоимости работы по разработке .....	47
4.6.1 Расчет затрат на оплату труда.....	47
4.6.2 Расчет затрат по социальному налогу.....	51
4.6.3 Расчет амортизационных отчислений.....	51
4.6.4 Расчет затрат на электроэнергию .....	52
4.6.5 Расчет накладных и прочих расходов.....	54
4.6.6 Расчет стоимости по всем статьям затрат и определение структуры затрат .....	55
4.7 Цена интеллектуального руда.....	56
4.8 Вывод.....	57
6 Безопасность жизнедеятельности .....	58
6.1 Анализ потенциально опасных вредных факторов в помещении при разработке. ....	58
6.2 Расчет системы вентиляции. ....	61
6.3 Расчет системы освещения.....	64
6.3.1 Расчет естественного освещения.....	64
6.3.2 Расчет искусственного освещения точечным методом. ....	68
Заключение.....	71
Список литературы.....	72

## **Введение**

Во времена технического прогресса и глобальной компьютеризации, такая тема, как влияние индустрии компьютерных игр на мировую экономику, является более чем актуальной. Мировые игровые издательства в наше время превосходят в экономическом потенциале многие устоявшиеся корпорации. Крупные игровые издательства нашли место и прочно осели в нише рынка, где конкурируют наравне с бизнес гигантами. В данном исследовании я обозначу понятие «индустрия компьютерных игр» и определю возможность дальнейшего развития игровой индустрии в мировой экономике.

Эта индустрия берёт своё начало в 1970-х годах, а прародителями её была горстка энтузиастов. Обычное увлечение за несколько десятилетий выросло от небольшого рынка до мейнстрима, так как в этой индустрии свободно уживались и большие компании, и совсем небольшие независимые группы разработчиков. Говоря о самом понятии индустрии компьютерных игр, я говорю о множестве аспектов, которые входят в неё. Под индустрией компьютерных игр я буду понимать совокупность различных компаний, сообществ и отдельных личностей, а так же технологий и процессов, которые вместе образуют полный цикл производства (разработка, продажа, продвижение, потребление) компьютерных игр. Так же это сектор экономики, который связан с разработкой, продажей и продвижением компьютерных игр, что позволяет работать в этой области огромному количеству людей с самыми различными специальностями от гейм-дизайнеров и художников до программистов и менеджеров. Одной из самых первых впоследствии разросшихся компаний являлась компания «Atari», которая оказала огромное влияние на зарождение игровой индустрии, как перспективной сферы экономики. Она была основана Ноланом Бушнеллом в 1972 и до 1984 распространилась по территории США и стала самой быстро растущей компанией страны. В те же 70-ые и 80-ые произошло очень много событий связанных с игровой индустрией. Это и многочисленные покупки компаний более успешными и большими компаниями, и возникновение десятков новых компаний, и первый кризис 1983 года, который повлёк спад максимально дохода от 3,2 млрд долларов до 100 млн долларов в 1985, что привело к прерыванию жизненного цикла второго поколения игровых консолей в Северной Америке. Кризис почти уничтожил индустрию и повлёк к банкротству множества компаний производителей домашних компьютеров и игровых приставок, в том числе «Atari». Одной из главных причин кризиса стало перенасыщение рынка огромным количеством игр весьма посредственного качества, что спровоцировало потерю потребительского доверия. Но кризис был преодолен благодаря работе таких компаний как «Atari» и «Nintendo». Вскоре игровая



индустрия начала набирать новые обороты, но в этот раз были приняты меры защиты индустрии от очередного кризиса. Игры вновь заняли свою нишу в экономике стран, но теперь более аккуратно располагая своими ресурсами.

В настоящее время трудно представить нашу планету без различных компьютеров, игровых консолей и подобной техники. Возможно игровая индустрия не является определяющей в вопросах экономики, но она безусловно играют огромную роль в её вопросах.

## 1 Особенности выбора технологий

### 1.1 Язык программирования C#

C# — элегантный, типобезопасный объектно-ориентированный язык, предназначенный для разработки разнообразных безопасных и мощных приложений, выполняемых в среде .NET Framework. С помощью языка C# можно создавать обычные приложения Windows, XML-веб-службы, распределенные компоненты, приложения "клиент-сервер", приложения баз данных и т. д. Visual C# предоставляет развитый редактор кода, конструкторы с удобным пользовательским интерфейсом, встроенный отладчик и множество других средств, упрощающих разработку приложений на базе языка C# и .NET Framework.

Синтаксис C# очень выразителен, но прост в изучении. Все, кто знаком с языками C, C++ или Java с легкостью узнают синтаксис с фигурными скобками, характерный для языка C#. Разработчики, знающие любой из этих языков, как правило, смогут добиться эффективной работы с языком C# за очень короткое время. Синтаксис C# делает проще то, что было сложно в C++, и обеспечивает мощные возможности, такие как типы значений Nullable, перечисления, делегаты, лямбда-выражения и прямой доступ к памяти, чего нет в Java. C# поддерживает универсальные методы и типы, обеспечивая более высокий уровень безопасности и производительности, а также итераторы, позволяющие при реализации коллекций классов определять собственное поведение итерации, которое может легко использоваться в клиентском коде. Выражения LINQ делают строго типизированный запрос очень удобной языковой конструкцией.

Как объектно-ориентированный язык, C# поддерживает понятия инкапсуляции, наследования и полиморфизма. Все переменные и методы, включая метод Main — точку входа приложения — инкапсулируются в определении классов. Класс может наследовать непосредственно из одного родительского класса, но может реализовывать любое число интерфейсов. Для методов, которые переопределяют виртуальные методы в родительском классе, необходимо ключевое слово `override`, чтобы исключить случайное повторное определение. В языке C# структура похожа на облегченный класс: это тип, распределяемый в стеке, реализующий интерфейсы, но не поддерживающий наследование.

В дополнение к основным описанным объектно-ориентированным принципам, язык C# упрощает разработку компонентов программного

обеспечения благодаря нескольким инновационным конструкциям языка, в число которых входят следующие:

- Инкапсулированные сигнатуры методов, называемые делегатами, которые поддерживают типобезопасные уведомления о событиях.
- Свойства, выступающие в роли методов доступа для закрытых переменных-членов.
- Атрибуты с декларативными метаданными о типах во время выполнения.
- Встроенные комментарии XML-документации.
- LINQ, предлагающий встроенные возможности запросов в различных источниках данных.

Процесс построения C# по сравнению с C и C++ прост и является более гибким, чем в Java. Нет отдельных файлов заголовка, а методы и типы не требуется объявлять в определенном порядке. В исходном файле C# может быть определено любое число классов, структур, интерфейсов и событий.

### **1.1.1 Роль платформы .NET**

Когда говорят C#, нередко имеют в виду технологии платформы .NET (WPF, ASP.NET). И, наоборот, когда говорят .NET, нередко имеют в виду C#. Однако, хотя эти понятия связаны, отождествлять их неверно. Язык C# был создан специально для работы с фреймворком .NET, однако само понятие .NET несколько шире.

Как-то Билл Гейтс сказал, что платформа .NET - это лучшее, что создала компания Microsoft. Возможно, он был прав. Фреймворк .NET представляет мощную платформу для создания приложений. Можно выделить следующие ее основные черты:

- Поддержка нескольких языков. Основой платформы является общезыковая среда исполнения Common Language Runtime (CLR), благодаря чему .NET поддерживает несколько языков: наряду с C# это также VB.NET, C++, F#, а также различные диалекты других языков, привязанные к .NET, например, Delphi.NET. При компиляции код на любом из этих языков компилируется в сборку на общем языке CIL (Common Intermediate Language) - своего рода ассемблер платформы .NET. Поэтому мы можем сделать отдельные модули одного приложения на отдельных языках.
- Кроссплатформенность. .NET является переносимой платформой (с некоторыми ограничениями). Например, последняя версия платформы на данный момент .NET Framework поддерживается на большинстве современных ОС Windows (Windows 10/8.1/8/7/Vista). А благодаря проекту Mono можно создавать

приложения, которые будут работать и на других ОС семейства Linux, в том числе на мобильных платформах Android и iOS.

– Мощная библиотека классов. .NET представляет единую для всех поддерживаемых языков библиотеку классов. И какое бы приложение мы не собирались писать на С# - текстовый редактор, чат или сложный веб-сайт - так или иначе мы задействуем библиотеку классов .NET.

Также еще следует отметить такую особенность языка С# и фреймворка .NET, как автоматическая сборка мусора. А это значит, что нам в большинстве случаев не придется, в отличие от С++, заботиться об освобождении памяти. Вышеупомянутая общезыковая среда CLR сама вызовет сборщик мусора и очистит память.

### **1.1.2 Архитектура платформы .NET Framework**

Программа на языке С# выполняется в среде .NET Framework – интегрированном компоненте Windows, содержащем виртуальную систему выполнения (среда CLR) и унифицированный набор библиотек классов. Среда CLR представляет собой коммерческую реализацию Майкрософт инфраструктуры CLI (common language infrastructure), международного стандарта, основы сред выполнения и разработки с тесным взаимодействием языков и библиотек.

Исходный код, написанный на языке С#, компилируется в промежуточный язык (IL) в соответствии со спецификацией CLI. Код IL и ресурсы, такие как растровые изображения и строки, хранятся на диске в исполняемом файле, называемом сборкой, с расширением EXE или DLL в большинстве случаев. Сборка содержит манифест со сведениями о типах сборки, версии, языке и региональных параметрах и требованиях безопасности.

При выполнении программы на С# сборка загружается в среду CLR в зависимости от сведений в манифесте. Далее, если требования безопасности соблюдены, среда CLR выполняет JIT-компиляцию для преобразования кода IL в инструкции машинного кода. Среда CLR также предоставляет другие службы, относящиеся к автоматическому сбору мусора, обработке исключений и управлению ресурсами. Код, выполняемый средой CLR, иногда называют "управляемым кодом" в противопоставление "неуправляемому коду", который компилируется в машинный код, предназначенный для определенной системы. Далее показаны отношения во время компиляции и время выполнения между файлами с исходным кодом С#, библиотеками классов .NET Framework, сборками и средой CLR.

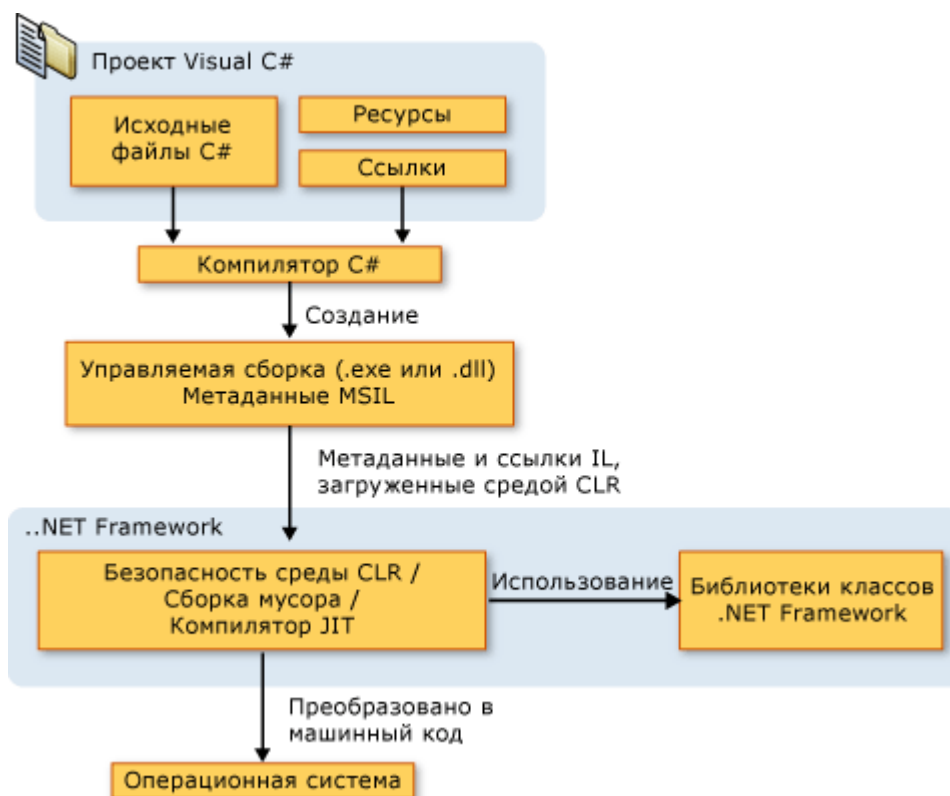


Рисунок 1.3 – отношения между файлами во время сборки и во время компиляции

Взаимодействие между языками является ключевой особенностью .NET Framework. Поскольку код на промежуточном языке (IL), создаваемый компилятором C#, соответствует спецификации CTS, код IL на основе C# может взаимодействовать с кодом, создаваемым версиями языков Visual Basic, Visual C++, Visual J# платформы .NET Framework и еще более чем 20 CTS-совместимых языков. В одной сборке может быть несколько модулей, написанных на разных языках платформы .NET Framework, и типы могут ссылаться друг на друга, как если бы они были написаны на одном языке.

Помимо служб времени выполнения, в .NET Framework также имеется обширная библиотека, состоящая из более чем 4000 классов, организованных по пространствам имен, которые обеспечивают разнообразные полезные функции для любых действий, начиная от ввода и вывода файлов для управлением строками для разбивки XML, и заканчивая элементами управления Windows Forms. В обычном приложении на языке C# библиотека классов .NET Framework интенсивно используется для "устройства" кода.

### 1.1.3 Управляемый и неуправляемый код

Нередко приложение, созданное на C#, называют управляемым кодом (managed code). Что это значит? А это значит, что данное приложение создано на основе платформы .NET и поэтому управляется общезыковой средой CLR,

которая загружает приложение и при необходимости очищает память. Но есть также приложения, например, созданные на языке C++, которые компилируются не в общий язык СIL, как C# или VB.NET, а в обычный машинный код. В этом случае .NET не управляет приложением.

#### **1.1.4 JIT-компиляция**

Как выше писалось, код на C# компилируется в приложения или сборки с расширениями exe или dll на языке СIL. Далее при запуске на выполнение подобного приложения происходит JIT-компиляция (Just-In-Time) в машинный код, который затем выполняется. При этом, поскольку наше приложение может быть большим и содержать кучу инструкций, в текущий момент времени будет компилироваться лишь та часть приложения, к которой непосредственно идет обращение. Если мы обратимся к другой части кода, то она будет скомпилирована из СIL в машинный код. При том уже скомпилированная часть приложения сохраняется до завершения работы программы. В итоге это повышает производительность.

### **1.2 Платформа Mono/Xamarin?**

Mono — проект по созданию полноценного воплощения системы .NET Framework на базе свободного программного обеспечения. Основным разработчиком проекта Mono — корпорация Xamarin, ранее Novell. Проект возглавляет Мигель де Икаса, известный разработчик, основатель проекта GNOME. Реализации Mono существуют для следующих операционных систем: Windows, Linux, BSD, Solaris, Mac OS X, Apple iOS, Wii и т.д.

Xamarin — это фреймворк для кроссплатформенной разработки мобильных приложений (iOS, Android, Windows Phone) с использованием языка C#. Идея очень простая. Пишется код на любимом языке, с применением всех привычных для вас языковых фишек вроде LINQ, лямбда-выражений, Generic и async. При этом остается полный доступ ко всем возможностям SDK платформы и родному механизму создания UI, получая на выходе приложение, которое, строго говоря, ничем не отличается от нативных и не уступает им в производительности.

Фреймворк состоит из нескольких основных частей:

- Xamarin.iOS — библиотека классов для C#, предоставляющая разработчику доступ к iOS SDK;
- Xamarin.Android — библиотека классов для C#, предоставляющая разработчику доступ к Android SDK;
- Компиляторы для iOS и Android;

- IDE Xamarin Studio;
- Плагин для Visual Studio.

Некоторое время назад достаточно широкую известность получили ряд фреймворков(например PhoneGap), которые предлагают разработку кроссплатформенных мобильных приложений на HTML5 с использованием JavaScript. Идея заключается в том, что приложение разрабатывается как обычный сайт для мобильных устройств с использованием соответствующих js-библиотек, например, JQuery Mobile. Затем все это упаковывается в некий контейнер, который для пользователя выглядит как нативное приложение. Минусы этих фреймворков очевидны: во-первых, вы не имеете доступа к нативным элементам UI. То есть даже если вы хотите использовать стандартную кнопку «Назад» для iPhone, вы должны ее нарисовать и сверстать. Во-вторых, вы получаете урезанный и обобщенный API для работы с платформой. Таким образом, те или иные фишки, присущие какой-то отдельной платформе будут вам недоступны. Ну и третье и самое важное — такое приложение физически запускается внутри браузера телефона.



Рисунок 1.4 – разделение кода для разных платформ

Xamarin основан на open-source реализации платформы .NET — Mono. Эта реализация включает в себя собственный компилятор C#, среду выполнения, а так же основные .NET библиотеки. Цель проекта — позволить запускать программы, написанные на C#, на операционных системах, отличных от

Windows — Unix-системах, Mac OS и других. Важно, что разработкой Xamarin занимаются те же люди, что и разработкой Mono.

## 1.2 Проект XNA/Monogame

Microsoft XNA — набор инструментов с управляемой средой времени выполнения (.NET), созданный Microsoft, облегчающий разработку и управление компьютерными играми. XNA стремится освободить разработку игр от написания «повторяющегося шаблонного кода» и объединить различные аспекты разработки игр в одной системе. Набор инструментов XNA был анонсирован 24 марта 2004 на Game Developers Conference в Сан-Хосе, Калифорния.

Из письма Microsoft от 31 января 2013 стало известно, что новые версии XNA более не будут разрабатываться и XNA не будет доступен в новом Metro интерфейсе Windows 8, а также на Windows 8 RT.

XNA Framework основывается на нативной реализации .NET Compact Framework 2.0 для разработки игр для Xbox 360 и .NET Framework 2.0 на Windows. Он включает обширный набор библиотек классов, специфичных для разработки игр, поддерживающий максимальное повторное использование кода на всех целевых платформах. Фреймворк выполняется на модификации Common Language Runtime, оптимизированной для игр, чтобы предоставить управляемую среду выполнения. Среда времени выполнения доступна для Windows XP, Windows Vista, Windows 7 и Xbox 360. Так как игры XNA пишутся для среды времени выполнения, они могут быть запущены на любой платформе, поддерживающей XNA Framework с минимальными изменениями или вообще без таковых. Игры, которые запускаются на фреймворке, технически могут быть написаны на любом .NET-совместимом языке, но официально поддерживается только язык программирования C# и среды быстрой разработки XNA Game Studio Express и все версии Visual Studio 2005.

XNA Framework скрывает низкоуровневые технологические детали, связанные с разработкой игры. Таким образом, фреймворк заботится о разнице между платформами, позволяя разработчикам уделять больше внимания смысловому содержанию игры. XNA Framework интегрируется с несколькими инструментами, такими как XACT, для помощи в создании контента. XNA Framework предоставляет поддержку создания и двухмерных, и трёхмерных игр и позволяет использовать возможности контроллеров Xbox 360. Игры фреймворка XNA, предназначенные для платформы Xbox на данный момент могут быть проданы только членам клуба Microsoft XNA Creator's Club.



Десктопные приложения могут распространяться бесплатно под текущим лицензированием Microsoft.

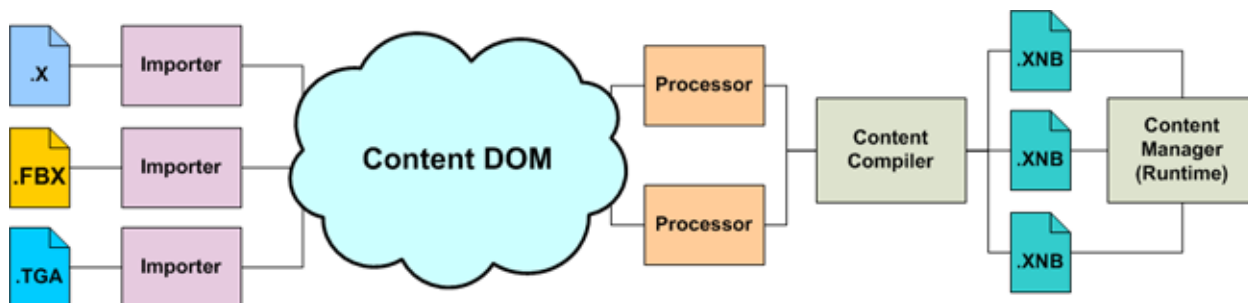


Рисунок 1.5 – схема работы Content Pipeline

XNA Content Pipeline — набор интегрированных в Visual Studio инструментов управления конвейером ресурсов игры, помогающий при определении, поддержке, отладке и оптимизации конвейера ресурсов игры. Конвейер ресурсов игры описывает процесс, при котором содержимое игры, такое как текстуры и трёхмерные модели, преобразуется в форму, подходящую для использования игровым движком. Content Pipeline помогает идентифицировать зависимости конвейера, а также предоставляет доступ к программному интерфейсу, позволяющему производить дальнейшую обработку данных о зависимостях. Данные о зависимостях могут быть проанализированы для уменьшения размера игры посредством выявления содержимого, которое не используется.

Все добавленные в такую папку (или проект) файлы расцениваются как отдельные единицы контента. И при выборе такого проекта, на панели Properties (Свойства) отдельно отображаются параметры, связанные с XNA Content Pipeline, в том числе назначенные модули для импорта и обработки. И для конкретного модуля обработки — свои специфические параметры.

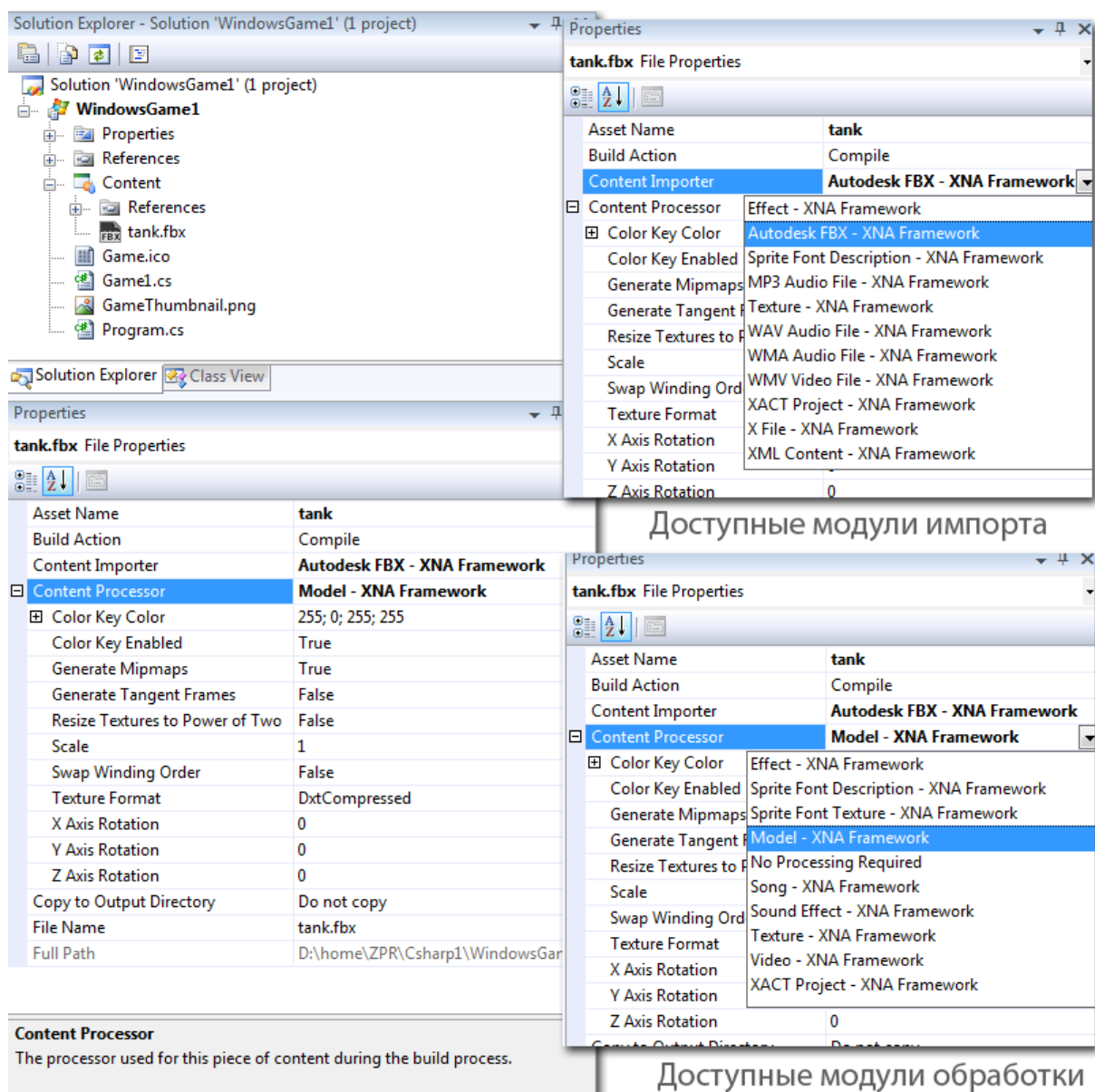


Рисунок 1.6 – сборка игровых ресурсов

При этом такая папка (или проект) имеет свой список «References», сборки из которого не только используются во время компиляции, но также используются как источник модулей (content importer, content processor) для конвейера. Когда возникает необходимость показать список доступных модулей, content pipeline сканирует сборки, добавленные в «References», на наличие классов-модулей конвейера (определяет по базовому классу или интерфейсу).

MonoGame представляет кроссплатформенную OpenSource-реализацию популярного фреймворка Microsoft XNA 4, который предназначен для работы с графикой и прежде всего для создания игр.

Microsoft XNA 4 позволял с помощью стандартных для платформы .NET языков C# и VB.NET относительно быстро и легко создавать сложные

визуальные сцены вплоть до трехмерных игр с богатой графикой и изоощренными игровыми сценариями. Правда, у XNA 4.0 был один недостаток - данная платформа, как и все продукты Microsoft в то время (2010-2011 годы) была нацелена исключительно на ОС Windows. Позже появилась возможность работать с XNA в рамках мобильной платформы для Windows Phone 7. Однако несмотря на популярность фреймворка в 2012 Microsoft официально прекратила его развитие.

Чуть ранее в 2009 году один из активных участников сообщества XNA Хосе Антонио Лил де Фариас (Jose Antonio Leal de Farias) запустил проект, который назывался XNA Touch. Его цель состояла в портировании простеньких 2D-игр на XNA на мобильные устройства, в частности на iPhone. И в декабре 2009 года была опубликована на Codeplex первая игра, которая поддерживала только iPhone. Впоследствии в AppStore появились десятки игр, использовавшие XNA Touch.

В марте 2011 года проект был переименован в MonoGame и переместился на GitHub, где и по сей день находится. В тот же год начал добавляться поддержка для платформ Android, Mac, Linux и OpenGL на Windows. Таким образом, проект стал уходить от ориентации на одну платформу, которая была присуще XNA.

В 2012 стала добавляться поддержка DirectX 11 и новой ОС Windows 8. Так, после закрытия проекта XNA платформа Monogame сразу успешно заняло нишу некогда популярного фреймворка в отрасли игростроения. И появилась возможность размещать игры на XNA в магазине Windows Store. Кроме Win8 Monogame начал поддерживать и новую мобильную платформу Windows Phone 8.

В 2013 году появились планы охватить также PlayStation 4.

В итоге к настоящему моменту MonoGame более менее поддерживает следующие платформы: Xbox 360, Windows (в том числе последнюю 10-ю версию), Windows Phone, iOS, Android, Mac OS X, Linux.

К настоящему моменту MonoGame практически полностью внедрил все функции, которые имелись в XNA 4, и кроме того, предлагают кучу возможностей, которые ранее не имелись в XNA. Долгая история MonoGame отражается в версиях фреймворка: текущей версией на данный момент (июнь 2016) является версия MonoGame 3.5.

## **2 Компьютерные игры**

В современном мире создание видеоигр является одним из наиболее крупных сегментов индустрии развлечений. Масштабы игровой индустрии сопоставимы, например, с киноиндустрией. А по скорости роста за последние пять лет индустрия видеоигр существенно ее опережала.

По степени влияния на потребителей и вовлеченности их в интерактивное окружение, предлагаемое видеоиграми, этот сегмент уже давно выделяется среди других видов развлечений.

Геймдев или разработку игр невозможно рассматривать обособленно от индустрии компьютерных игр в целом. Непосредственно создание игр – это только часть комплексной «экосистемы», обеспечивающей полный жизненный цикл производства, распространения и потребления таких сложных продуктов, как компьютерные игры.

В структуре современной игровой индустрии можно выделить следующие уровни: платформы, игровые движки, разработка видеоигр, издание и оперирование, популяризация и потребление.

Очень небольшое количество учебных заведений в мире готовят специалистов непосредственно для игровой индустрии. В Казахстане же пока таких заведений нет и это отличная возможность занять нишу.

### **2.1 Игровые платформы**

Аппаратно-программные системы, позволяющие запускать интерактивные игровые приложения. Среди основных видов можно выделить:

- Персональные компьютеры на базе Windows, Mac/OS X или Linux
- Игровые консоли (специализированные устройства для игр, сейчас в ходу 8-е поколение – Xbox One, PlayStation 4, Nintendo Wii U)
- Мобильные устройства (iOS, Android, Windows)
- Универсальные web платформы, социальные сети (FB, VK, MM)
- Аркадные автоматы
- Инновационные платформы виртуальной реальности

### **2.2 Игровые движки**

Программная прослойка между платформой и собственно кодом игры. Использование готового игрового движка позволяет существенно упростить разработку новых игр, удешевить их производство и существенно сократить

время до запуска. Также современные игровые движки обеспечивают кроссплатформенность создаваемых продуктов. Из наиболее продвинутых движков можно выделить: Unity 3D, Unreal Development Kit, CryENGINE 3.

### **2.3 Разработка игр**

Большое количество компаний и независимых команд занимаются созданием компьютерных игр. В разработке участвуют специалисты разных профессий: программисты, гейм-дизайнеры, художники, QA специалисты и др.

К разработке крупных коммерческих игровых продуктов привлекаются большие профессиональные команды в 100+ специалистов. И стоит подобные проекты в разработке могут десятки миллионов долларов.

Однако вполне успешные игровые проекты могут воплощаться и небольшими командами энтузиастов. Этому способствует присутствие на рынке большого количества открытых и распространенных платформ, качественные и практически бесплатные движки, площадки по привлечению «народных» инвестиций (краудфандинг) и доступные каналы распространения.

### **2.4 Издание и оперирование игр**

Распространением игр или оперированием (в случае с ММО) занимаются, как правило, не сами разработчики, а издатели. При этом издатели (или операторы) локализуют игры, взаимодействуют с владельцами платформ, проводят маркетинговые компании, разворачивают инфраструктуру, обеспечивают техническую и информационную поддержку выпускаемым играм.

Для средних и небольших игровых продуктов данный уровень практически не доступен. Такие продукты, как правило, сами разработчики выводят на рынок, напрямую взаимодействуя с платформами.

### **2.5 Популяризация (маркетинг игр)**

Специализированные средства массовой информации всегда являлись мощным каналом донесения информации до пользователей.

Сейчас наиболее эффективным и широко представленным направлением СМИ являются информационные сайты, посвященные игровой тематике. Игровые журналы, долгое время выступавшие главным источником информации об играх, в настоящее время уступили свое место интернет ресурсам.

Специализированные выставки все еще остаются важным информационными площадками для игровой индустрии (E3, GDC, Gamescom, ИгроМир, КРИ, DevGamm). Прямое общение прессы и игроков с разработчиками, обмен опытом между участниками рынка, новые контакты - вот то, что предлагают конференции и выставки в концентрированной форме.

Еще один важный канал донесения полезной информации до игроков – это ТВ-передачи, идущие как в формате классического телевидения, так и на множестве каналов видео-контента.

## **2.6 Игровые сообщества**

В рамках наиболее популярных игровых направлений, вокруг конкретных игр или серий образуются игровые сообщества, возникают новые сайты, форумы и группы в социальных сетях. Зачастую подобные группы генерируют даже больший поток информации, чем могут себе позволить разработчики и издатели через собственные ресурсы и рекламу. Достаточно типично, когда независимый сайт по игре предлагает посетителям более полную информацию по миру игры, по сравнению с официальным сайтом.

## **2.7 Киберспорт**

В связи с широким распространением киберспорта наиболее опытные игроки получают возможность переводить свои увлечения играми на профессиональные рельсы. В мире проводится достаточно много чемпионатов, а ставки столь высоки, что результативные игроки могут сделать на этом хорошую карьеру киберспортсмена. Сейчас киберспорт – это целая индустрия со своей инфраструктурой, финансированием и знаменитостями.

## **2.8 Творчество игроков**

Наиболее преданные фанаты игр не только проводят за творениями игродела многие годы своей жизни, но и зачастую создают разнообразный контент вокруг любимой игры. Это могут целые сайты, рисунки, косплеи, журналы, видеопередачи, игровые дополнения или даже полноценные игры.

## **2.9 Индустрия компьютерных игр: история зарождения и развития**

Первые компьютерные игры появились еще на заре компьютерной эры, во времена ламповых ЭВМ. В 1952 году А. С. Дуглас как часть своей

университетской диссертации создал программу «ОХО», работавшую на большом компьютере EDSAC в Кембриджском университете. Это была простейшая реализация игры, известной всем как «крестики-нолики».

Но до тех пор, пока технологии не позволяли выпускать компактные и надежные компьютерные системы, создание игр было делом отдельных увлеченных научных сотрудников.

Началом же индустрии видеоигр принято считать запуск в 1971 году игры Computer Space на аркадных автоматах. Следующим шагом на пути становления индустрии видеоигр стал коммерчески успешный выпуск компанией Atari видеоигры Pong в 1972. Всего было продано 19 тыс. аркадных автоматов с этой игрой.

В том же году на рынке появилась первая домашняя игровая консоль Magnavox Odyssey.

По мере развития технологий и появления новых (поначалу специализированных, а затем и универсальных) устройств, индустрия видеоигр тоже развивалась и адаптировалась к новым реалиям.

Конец 1970-х и начало 1980-х стали «золотой эрой аркадных автоматов» автоматов.

Среди наиболее значимых игр данного периода можно отметить Space Invaders (1978) и PacMan (1980). Новые игры уже существенно отличались в лучшую сторону от родоначальников индустрии визуально и в плане игрового процесса.

Суммарный мировой доход игровой индустрии в 1982 году вплотную приблизился к \$12 млрд. (или почти \$30 млрд. в ценах 2012 года).

В 1980-е с появлением персональных компьютеров начало формироваться направление разработки игр для них. Поначалу игры создавали энтузиасты-разработчики, но по мере роста производительности и распространения персональных систем, для них стали выпускаться уже коммерческие игровые продукты. Игры для персональных систем существенно отличались от классических аркадных. В это время появились первые стратегии и сложные RPG миры. Но все же основой игровой индустрии оставались аркадные автоматы и специализированные игровые приставки.

В это время появились первые печатные издания, посвященные компьютерным играм.

В 1990-е дальнейшее развитие цифровых технологий (внедрение CD-ROM, развитие аппаратных 3D технологии визуализации, появление интернета) и распространение GUI ориентированных операционных систем (Microsoft и Mac OS) привели к существенному увеличению доли компьютерных игр.

Именно в этот период появляются все основные жанры и возникает большое разнообразие игровых механик.

Становится доступной совместная игра в клубах или через интернет. Зарождается киберспорт.

В середине 1990-х появляется первое учебное заведение, готовящее специалистов по разработке видео игр. DigiPen Institute of Technology открывается в Ванкувере в 1994 году.

В 2000-е основными движущими силами развития игровой индустрии становится широкое распространение интернета и появление достаточно производительных для реализации игр мобильных устройств.

В 2007 году выходит iOS, а Facebook предлагает сторонним разработчикам создавать приложения на своей социальной платформе. В 2008 году Google выпускает на рынок первую версию Android с полноценным пакетом разработчика SDK 1.0.

Наиболее значительным событием этого периода становится выход игры World of Warcraft в 2004 году. Это самая популярная и самая долго живущая ММО игра в мире. Фактически она определила направление развития этого сегмента игровой индустрии на десятилетия вперед. На декабрь 2014 года в игре было 10 млн. подписчиков.

В этот период развитие киберспорта достигло небывалых высот. Интересный факт. Спустя несколько лет после выхода игра Star Craft в Южной Корее стала национальным видом спорта, собирая целые стадионы на финальные матчи. В период наибольшей популярности ПРО геймеры зарабатывали от \$20 до \$600 тыс. в год.

В 2003 году появляется уникальное образовательное учреждение (The Guildhall at SMU, штат Техас), обучающее специальностям геймдева, в частности геймдизайну.

Уникально оно тем, что в разработке учебных программ принимали активное участие легенды мирового игростроя Том Холл, Джон Ромеро, Пауль Джаквейс, Тим Виллитс и Ричард Бейли Грей aka Levelord! А учебные корпуса напоминали здания из компьютерной игры.

В дальнейшем уже во многих странах начинает развиваться gamedev образование.

## **2.10 Современность и прогнозы**



В предыдущие пять лет игровая индустрия во всем мире переживает бурный рост. Рост наблюдался во всех сегментах, но основными драйверами в этот период стали мобильные игры на двух основных платформах iOS и Android.

Стоит отметить, что наибольший вклад в объемы индустрии игр вносят США и Китай, но структура рынка у этих фаворитов сильно различается. В США самый крупный сегмент – это консольные игры, а Китае – ММО.

Интересна причина такой специфичности китайского рынка игр. До 2013 года в Китае существовал мораторий на продажу игровых консолей. Этот запрет стал катализатором быстрого развития направления ММО игр. В результате Китай стал основным производителем огромного количества зачастую очень похожих друг на друга, но в тоже время достаточно качественных игровых проектов ММО сегмента. Игровой рынок Китая в 2014 оценивался почти в \$18 млрд., а количество геймеров в 173 млн.

Среди самых значительных событий в 2014 году можно отменить:

Покупка компанией Microsoft игры Minecraft за \$2.5 млрд. Изначально это был независимый проект программиста-одиночки. Но по мере развития он завоевал армию поклонников. Будучи построенным на открытой технологии, он получил взрывной рост по наполнению и обошел по популярности все остальные развлечения среди детей цифрового поколения. На осень 2014 года количество зарегистрированных игроков превысило 10 млн, причем это произошло абсолютно без привлечения рекламы.

Покупка Facebook технологии Oculus VR за \$2 млрд. Понятно, что технологию можно использовать во множестве областей и для различных задач, но для игровой индустрии – эта технология стала принципиальным прорывом для формирования нового VR направления.



Рисунок 2.1 - Объем мирового рынка

## 2.11 Игры как искусство

Современные компьютерные игры врываются в сопредельные общественные и культурные сферы – искусство, образование, этику, психологию, социальные коммуникации и даже спорт (в мире давно проводятся полноценные киберспортивные чемпионаты с солидными бюджетами. При всей очевидности их технологической составляющей, компьютерные игры – продукт, в значительной степени синтетический. Как верно отмечает Д.В. Галкин, «наравне с компьютерной графикой и web-дизайном, компьютерные игры являются техно-художественными гибридами, в которых технологическая основа служит не только инструментом создания художественного продукта, но включена в художественное содержание и эстетические свойства произведения».

В этой связи возникает очевидный вопрос: можно назвать современные компьютерные игры искусством, или это просто, пусть и высокотехнологичная, но форма развлечения? Действительно, в СМИ, в специализированных компьютерных изданиях, в ряде аналитических статей на Западе и в России компьютерные игры чаще всего именуется продуктом целой индустрии электронных развлечений, а не произведениями искусства. На мой взгляд, в этом «виновата» их и новая, еще не до конца осмысленная мультимедийная форма, и массовость, и установка на коммерцию и собственно развлечение, совершенно нетрансцендентное (на первый взгляд) времяпрепровождение. Кроме того, отдельной проблемой для исследователя является то, что в большинстве источников, касающихся темы

компьютерных игр (а это, в первую очередь материалы СМИ) границы искусства сужаются, т.е. искусство как таковое понимается (в романтическом ключе) как нечто элитарное, арт-хаусное, что в целом, характерно для массового сознания и научно-популярных работ. Исходя из этого, компьютерные игры, с их первоочередной направленностью на развлечение таким искусством считаться не может. Разве способствуют они познанию, пробуждению? Восприятию трансцендентных истин и чувственно-имманентному переживанию катарсиса?

В этом контексте более верно отнести компьютерные игры (большую их часть) в сферу массовой, поп-культуры, где главное – не тонкие эстетические переживания, а, прежде всего, коммерческая выгода. Именно поэтому, в большинстве современных компьютерных игр (как и в блокбастерном кинематографе) основной акцент делается не на глубокое внутреннее содержание, а на развлечение, примитивный сюжет (затрагивающий основные человеческие чувства, базирующиеся на животных инстинктах), внешнюю яркость и аддиктивность, спецэффекты. Массовый человек не хочет вникать во внутреннее содержание, ему достаточно привлекательной обертки. Конечно, создаются и действительно многоплановые, сложные, художественные работы рассчитанные на аудиторию с должным уровнем эстетического образования. Но, чаще всего, такие проекты имеют небольшой бюджет и малоизвестны широкой публике.

Однако, несмотря на всю коммерческую направленность, остается вопрос: может ли быть шанс у компьютерной игры стать в один ряд с известными произведениями традиционных искусств? Ведь создатели компьютерных игр часто говорят, что хотели создать и дать аудитории нечто большее, чем просто развлечение на несколько часов. Интересно, что, несмотря на то, что у блокбастерных игр, как правило, нет автора, они – плод коллективной работы группы разработчиков, где каждый отвечает за свое направление (дизайн уровней, разработка engine, создание моделей NPC, бета-тестирование и т. д.), появляются и в полной мере «авторские» проекты, которых важнейшую роль играет выражение творческой индивидуальности автора. В таких работах ставка делается не на максимальные результаты в первую неделю продаж, но преследуются, в первую очередь, неутилитарные, эстетические цели – такие как создание нетривиального художественного образа. Таким примером может быть культовая «American McGee's Alice».

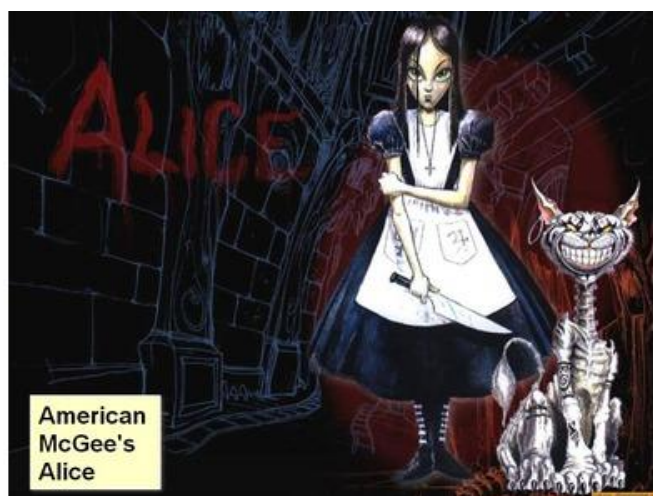


Рисунок 2.1 – Игра «Alice American McGee»

В этой игре разработчик American McGee не только создает виртуальное развлечение, но (что отражено в названии), предлагает свое авторское видение произведения Л. Кэрролла, трансформированного в игре в фантасмагоричную визионерскую реальность.

Для искусства важна передача информации посредством художественных образов. А художественный образ способен перевести реципиента с уровня повседневного дискурсивного мышления на более высокий уровень эстетического восприятия. В этой связи интересно посмотреть, что говорят сами разработчики, авторы концепций компьютерных игр о своих проектах.

В качестве примера обращусь к такой нетривиальной игре как «Spore».



Рисунок 2.3 - Игра «Spore» (выбор цвета)

Игра начинается со своеобразного демиургического конструктора – когда из отдельных элементов и форм (глаз, ушей, ног, рук, зубов, когте и т.п.) можно создать абсолютно любое неантропоморфное существо. Примечательно, что когда игра только начала издаваться, в сети был объявлен конкурс на самое интересное существо. И как вы думаете какое существо оказалось самым популярным? Чебурашка? Нет, фантазия пользователей двинулась в одном «магистральном» направлении - интернет заполнили миллионы шагающих, прыгающих, ползающих и летающих фаллосов – и в этом, заметны и очевидные фрейдистские мотивы, и реактуализация архаики, и штампы массовой культуры.

Но вот что пишет разработчик Уилл Райт о «Spore»: «в глобальном смысле я бы хотел, чтобы люди, играя в Spore, посмотрели на жизнь в перспективе. Чтобы они задумались о масштабе пространства и времени. Чтобы они хотя бы на секунду поняли, сколь долго существует жизнь на нашей планете. Spore дает почувствовать этот момент – как из микроскопического белкового соединения вырастают цивилизации. Вообще, Spore – это игра о важности жизни».

Интересно также, что в процессе разработки авторы постоянно переосмысливают свои произведения не только с утилитарной, но и с эстетической точки зрения. Тот же Уилл Райт, посвятивший всю жизнь созданию различных симуляторов творчески «эволюционировал» от первоначальной идеи застройки города к «симулятору всего», как часто называют «Spore» в игровой прессе.



Рисунок 2.4 - Игра «Spore» (выбор частей тела)

Как и в традиционных изобразительных искусствах (например, в живописи, от набросков – к итоговой картине) можно проследить творческие

метания автора, который словно ведет нас от зарождения мысли, эмоции к ее перерождению, трансформации. В то же время, игровой опыт обнаруживает признаки духовного эстетического акта: трансцендируя линейность времени пространства (поставив игру – время на паузу и отмотав картину на максимальный масштаб), мы словно замираем, пораженные величиим созданной вселенной, создается ощущение, что мы с автором переживаем весь его духовный путь, испытываем подлинное эстетическое переживание, связанное с переходом в состояние полноты бытия – контакта с Универсумом. При этом, остается дискуссионным вопрос, насколько эти переживания действительно сопоставимы с эстетическими переживаниями от традиционных произведений искусства. В значительной степени это вопрос личного эстетического воспитания и образования, а также знакомства с современными феноменами медиа-искусства и электронной культуры.

### **3. Разработка компьютерных игр**

Дизайн-документ — это детальное описание разрабатываемой компьютерной игры. Диздок создается и редактируется командой разработчиков и в основном используется в индустрии видеоигр для организации работы разработчиков. Документ создается в результате сотрудничества между дизайнерами, художниками и программистами как руководство, которое используется в процессе разработки. Когда издатель поручает создание игры разработчикам, команда разработчиков должна создать документ, который часто связан с соглашением между издателем и разработчиком; разработчики должны придерживаться дизайн-документа во время процесса формирования игры.

Разработчики игры могут представить дизайн-документ на подготовительной стадии разработки игры — до или после принятия решения по финансированию. Перед подачей документ может быть концептуальным и не полным. Как только проект одобрен, документ расширяется разработчиком до уровня, когда им может успешно руководствоваться команда разработчиков. В связи с динамическими условиями разработки игр, документ часто изменяется, пересматривается и расширяется по мере продвижения разработки, изменений в сфере и изучения направления. Также дизайн-документ часто называют живым документом, то есть таким, который постоянно совершенствуется и дописывается на протяжении всего периода воплощения проекта (иногда даже ежедневно). В начале документ может представлять собой лишь базовую концепцию проекта и стать полным подробным описанием каждого аспекта игры к концу разработки.

Дизайн-документ может содержать текст, изображения, диаграммы, концепт-арты или любые другие данные, необходимые для того, чтобы лучше проиллюстрировать проектные решения. Некоторые дизайн-документы могут включать в себя функционирующие прототипы или выбранный игровой движок. Хотя принимаются во внимание требования многих компаний, у дизайн-документа нет стандартной формы набора. Например разработчики могут решить хранить его как обработанный текстовый документ, или как онлайн-инструмент для совместной работы.

Цель дизайн-документа заключается в том, чтобы однозначно описать коммерческие аспекты игры, целевую аудиторию, игровой процесс, графику, дизайн уровней, историю (сюжет), персонажей, пользовательский интерфейс, средства и т. д. Короче говоря, для осуществления поставленной цели, каждое требование к разработке определенной части игры должно быть достаточно подробно описано для соответствующих разработчиков (дизайнеров, программистов и т. д.). Документ намеренно разделен таким образом, чтобы разработчики игры могли поддерживать отдельные его части.

Большинство видеоигр требуют включения следующих разделов:

- Игровой процесс (геймплей)
- История
- Персонажи
- Дизайн уровней/окружающей среды
- Концепт-арты
- Звук и музыка
- пользовательский интерфейс, управление

Этот список ни в коем случае не является исчерпывающим или применимым к каждой игре. Некоторые из этих разделов могут не присутствовать непосредственно в дизайн документе, но появятся в сопутствующей документации.

### **3.1 Предпроизводственный процесс**

Обычно перед началом разработки любой игры должна сформироваться идея, а издатель/разработчик должен дать «зелёный свет».

В более распространённом случае, если разработчик и издатель являются разными компаниями, идея должна быть предложена руководству, одобрена и выставлена на рассмотрение издателям. В этом деле может помочь рабочее демо, но оно не является обязательным для авторитетного издателя с хорошей репутацией. Если заинтересованный издатель найден, можно начинать

производство. Сегодня идея игры редко убеждает, если в ней не заинтересован издатель.

Если разработчик также является издателем, или если они оба являются подразделениями одной компании, то одобрение должно дать только высшее руководство. Однако, в зависимости от размера компании-издателя, может потребоваться несколько попыток, пока идея не поднимется вверх через все слои руководства.

Представителем проекта обычно является геймдизайнер, но им также может быть человек из игровой индустрии любой другой должности. Перед началом полномасштабного производства геймдизайнер должен написать дизайн документ — подробный документ, описывающий концепцию и геймплей. Также он может содержать некоторые предварительные скетчи (эскизы, рисунки) различных аспектов игры. Некоторые геймдизайнеры включают в дизайн-документ даже примерный рабочий прототип, демонстрирующий одну или несколько сторон игры. Обычно дизайн-документ объединяет в себе все или большую часть материалов начального замысла. Основная особенность дизайн-документа — это его «живость» — в действительности он не будет завершён до тех пор, пока игра находится в разработке. Он может изменяться каждую неделю, иногда — каждый день. Поэтому, даже если дизайн-документ должен существовать в некоторой форме перед началом полномасштабного производства, он почти никогда не является завершённым дизайном, хотя может описывать многие аспекты всех стадий полностью спроектированной игры.

Перед тем, как появится одобренный дизайн, основная команда программистов и художников может начать работу над идеями. Программисты могут разработать начальные прототипы для демонстрации одной или нескольких возможностей, которые хотят видеть в игре некоторые посредники. Или они могут начать разработку каркаса, который, в конечном счёте, будет использоваться игрой. Художники могут нарисовать эскизы, как плацдарм для разработки реальных игровых ресурсов. Сначала продюсер может работать над игрой неполный рабочий день, но повышать свою занятость по мере продвижения разработки.

### **3.2 Производство**

На этапе основного производства выполняется огромный объём работ. Программисты пишут исходный код, художники рисуют графику (спрайты или 3D-модели игровых элементов). Звукооператоры разрабатывают звуковые эффекты, а композиторы пишут музыку для игры. Дизайнеры уровней создают



уровни, а писатели пишут диалоги для скриптовых сцен и неигровых персонажей.

Всё это время геймдизайнер дополняет и изменяет игровой дизайн, чтобы отразить текущее видение игры. Некоторые особенности или уровни могут быть удалены, некоторые добавлены. Художественная трактовка может эволюционировать, а сюжет (предыстория) — измениться. Может появиться новая целевая платформа, а также новая целевая аудитория. Все эти изменения должны быть задокументированы и большинство из них должно появиться в дизайн-документе.

С точки зрения времени, первый уровень игры разрабатывается дольше всех остальных. Поскольку дизайнеры уровней и художники используют инструменты для создания уровней, им требуются возможности и изменения внутренних инструментов. С появлением новых возможностей некоторые уровни могут устареть, поэтому, в первый уровень игры могут вноситься различные исправления. Кроме того, в силу динамической природы разработки игр, дизайнерское видение первого уровня с течением времени может изменяться. Довольно обычным является потратить на первый уровень более 12 месяцев при общей трёхлетней разработке игры. Последующие уровни могут разрабатываться значительно быстрее, так как список возможностей становится более полным, а видение игры — более ясным.

Тестеры подключаются к игре, когда появляется что-то «играбельное». Это может быть один уровень или подмножество игры, которое может использоваться в любых разумных пределах. На раннем этапе тестирование игры отнимает у одного тестера относительно малую долю времени; в любое время тестеры могут быть ответственны сразу за несколько игр. По мере приближения разработки к концу, одна игра может начать отнимать у тестеров всё их время — и даже сверхурочно — поскольку они стараются протестировать новые возможности, для которых существуют регрессионные тесты. Сегодня тестирование является жизненно важным для игр, поскольку, в силу сложности большинства из них, одно-единственное изменение может привести к катастрофическим последствиям.

### **3.3 Поддержка**

В обычном случае поддержка заключается в выпуске патчей для исправления ошибок, найденных уже после выхода игры. Однако, в случае массовых многопользовательских онлайн-игр (ММО), поддержка может сравняться или даже превосходить производство как по трудоёмкости, так и по

времени, так как успешная ММО должна непрерывно развиваться и расширяться, чтобы избежать оттока игроков.

### **3.4 Игровые профессии**

В процессе производства игры в дело вступают представители множества "игровых" профессий – поговорим об основных из них. Кстати, при разработке небольшого проекта в рамках ограниченного бюджета один человек может совмещать в себе обязанности целой команды разработчиков.

#### **3.4.1 Программисты**

Программисты заняты работой по написанию программного кода игры. Их усилиями реализуется игровая физика, искусственный интеллект, с которым предстоит сражаться игроку при игре "против компьютера" и многое другое. Что интересно, многие игровые программисты стали таковыми после того, как начали программировать самостоятельно, в виде хобби.

Если говорить об инструментах программиста – то практически все коммерческие игры написаны на языке C++ или C, некоторые, особенно ответственные части игры, пишут на языке Assembler. В последнее время набирает популярность язык программирования C#. Строго говоря, игру реально написать на практически любом языке программирования – например, простые игры можно создавать в Microsoft Word или Microsoft Excel, используя встроенный в эти продукты Visual Basic For Applications.

#### **3.4.2 Художники**

Роль художников и вообще всех, кто работает с графикой, в современном игростроении трудно переоценить. Во все времена одним из критериев оценки игры была ее графическая составляющая – а современные средства работы с графикой позволяют создавать красивейшие игровые миры, в основе которых лежит кропотливая работа художников во все ее проявлениях.

К тому же, аниматоры оснащены сегодня передовым аппаратным и программным обеспечением (в частности – технологией Motion Capture) которое, например, позволяет записывать реальные движения человека и переносить их потом на игровых персонажей.

#### **3.4.3 Музыканты**

Композиторы, музыканты, актеры, звукорежиссеры работают над звуковым оформлением игры. Они пишут и исполняют музыку, читают тексты персонажей. Без достойной музыки и качественного озвучивания, как и без хорошей графики, современная игра вряд ли будет успешной. Хотя, звуки и музыка, конечно, не главное в большинстве игр, но, например, поиграйте в Need For Speed Undercover для Windows Phone 7 без звуков и музыки – немалая часть обаяния игры исчезнет без следа.

#### **3.4.4 Писатели**

Как правило, редкая современная игра не построена вокруг какого-то сценария. Особенно сильна роль сценариев в ролевых, приключенческих играх и в играх смешанных жанров, однако даже если работа ведется над очередной FPS-игрой – качественные тексты ей не помешают. Всем этим занимаются сценаристы, режиссеры, писатели.

#### **3.4.5 Дизайнеры уровней**

Когда программисты, художники, музыканты выполняют основную работу по созданию персонажей игры, игровых интерьеров, звуков, за дело берутся дизайнеры уровней. Если даже игра представляет собой один большой "уровень" без явного разбиения на части, дизайнерам приходится немало поработать над созданием игровой вселенной.

#### **3.4.6 Тестеры**

Игра – это огромная программа, которая перед выходом в свет должна быть протестирована. Тестирование занимает очень важную роль в процессе создания игр – иногда тестеров набирают из числа добровольцев. Желających поучаствовать в предварительном тестировании игры, как правило, больше, чем нужно – многие хотят попробовать новинку первыми. Если в ходе тестирования возникают ошибки, тестер сообщает о них разработчикам.

### **3.5 Перспективы программиста компьютерных игр**

Многие начинающие разработчики, которые хотят сами разрабатывать игры, задаются вопросом: "Может ли один человек, или, в крайнем случае, команда из нескольких энтузиастов, создать сегодня игру, которая принесет разработчикам славу и богатство и станет желанным гостем на игровых устройствах игроков всего мира"? Это сложный вопрос.

История знает множество примеров, когда у истоков какой-либо игры, ставшей безумно популярной во всем мире, стояла небольшая группа энтузиастов. Например – это всем известный Tetris, который в практически неизменном виде существует уже не одно десятилетие. Или игровая серия Final Fantasy – первая версия этой игры появилась в конце 1980-х годов, а теперь – это целая игровая вселенная, над новыми версиями которой работает огромное количество людей.

Если говорить о мобильных играх, то они, с одной стороны, отличаются всеми признаками современных игр. Серьезный мобильный проект требует не меньше усилий, чем игры для других платформ. Однако, особенности мобильных игр, в частности, высокая популярность достаточно простых по своему устройству игр, в которые играют, чтобы скоротать время (так называемых казуальных игр), открывает дорогу в мир мобильной разработки небольшим командам разработчиков. Такие игры сравнительно просты в разработке. Возьмем, например, популярнейшую игру Tetris, с разработкой собственной реализации которой справится и начинающий программист, различные варианты игры Lines, пасьянсы, бесчисленные реализации игр Snake, Pacman, Breakout. Эти игровые идеи живут уже многие годы (а то и десятилетия), и не теряют актуальности, постоянно перевоплощаясь, но сохраняя сущность исходных идей.

## **4 Экономическое обоснование проекта**

### **4.1 Описание работы и обоснование необходимости**

В настоящее время трудно представить нашу планету без различных компьютеров, игровых консолей и подобной техники. Возможно игровая индустрия не является определяющей в вопросах экономики, но она безусловно играют огромную роль в её вопросах. Объем мирового рынка представлен на рисунке 4.1.

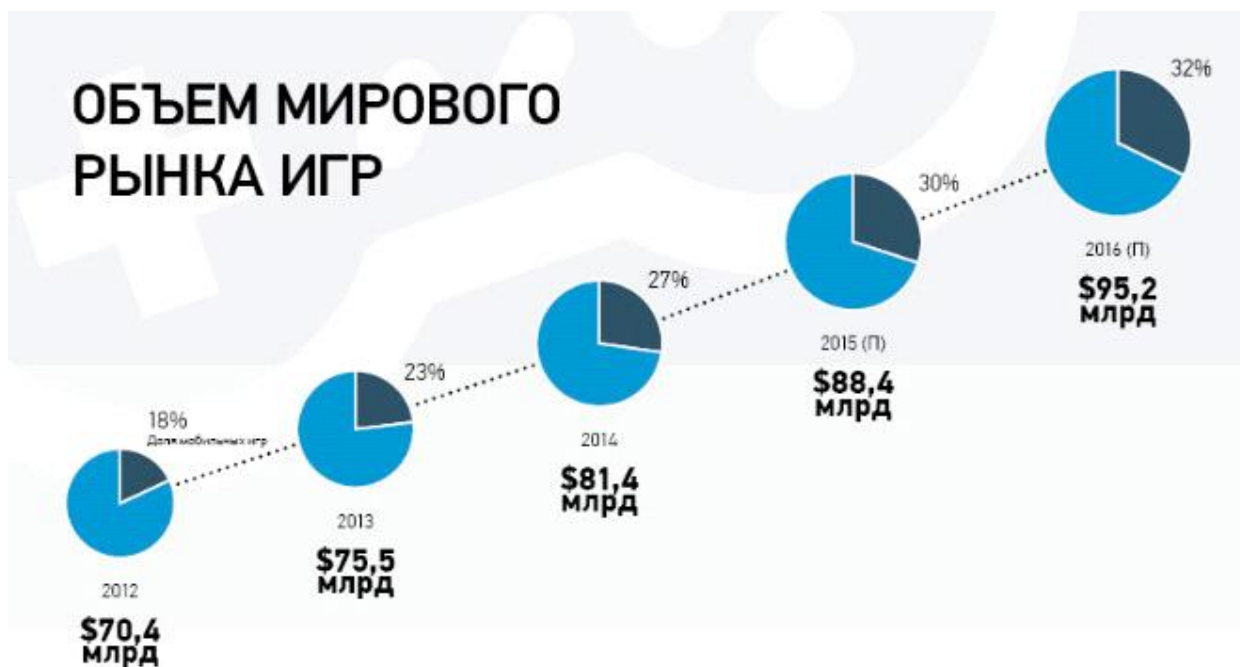


Рисунок 4.1 - объём мирового рынка игр.

Обращая внимание на данный рисунок можно с лёгкостью определить, что игровой рынок занимает очень большую часть рынка и продолжает стремительно расти. В настоящее время компьютерные игры вносят значительный вклад в мировую экономику ввиду большого успеха продаж основных игровых систем и игр вроде «Call of Duty: Black Ops» заработавшая в течение первых 5 дней продаж более 600 млн \$, что стало мировым рекордом пятидневных продаж среди фильмов, книг и компьютерных игр.

На сайте «NewZoo.com» приведен отчёт «Global Games Market Report», который содержит в себе информацию по валовому обороту игровой индустрии в разных странах за 2014 год. Отчёт составлен приблизительно, так как все аспекты индустрии учесть сложно, но цифры максимально приближены к реальности.

Таблица 1-валовый оборот игровой индустрии в разных странах за 2014.

Ранг	Страна	Численность населения страны	% населения в интернете	Количество интернет пользователей	Продаж на 1 человека, \$	Валовый объём продаж за год, \$

1	США	322 583 006	85,03	274 292 330	74,68	20 484 628 000
2	Китай	1 393 783 836	50,30	701 073 270	25,48	17 866 677 000
3	Япония	126 999 808	80,05	101 663 346	120,20	12 219 552 000
4	Германия	82 652 256	86,00	71 080 940	49,64	3 528 196 000
5	Великобритан ия	63 489 234	87,38	55 476 893	61,76	3 426 259 000
6	Республика Корея	49 512 026	86,10	42 629 854	78,73	3 356 202 000
7	Франция	64 641 279	87,00	56 237 913	46,39	2 608 818 000
8	Канада	35 524 732	89,77	31 890 552	53,87	1 717 991 000
9	Италия	61 070 224	60,40	36 886 415	41,05	1 514 067 000
10	Испания	47 066 402	79,40	37 370 723	39,85	1 489 366 000
11	Бразилия	202 033 670	58,70	118 593 764	11,29	1 339 375 000
12	Российская Федерация	142 467 651	61,81	88 059 255	12,98	1 143 197 000

Очевидно, что США лидер в этой индустрии. Страна действительно имеет мощный потенциал в индустрии компьютерных игр и она является неотъемлемой частью экономики данной страны. Такие всемирные известные американские компании как «Microsoft» и «Apple» в 2011 году входили в двадцатку самых крупнейших компаний США по версии журнала «Forbes». Так же говоря об успешности таких компаний можно упомянуть ещё одну американскую компанию «Valve». «Valve» содержит свой сервис цифрового распространения компьютерных игр и программ под названием «Steam».

По данным журнала «Форбс» на момент 2011 года аудитория магазина «Steam» — 30 млн человек. По состоянию на 2011 год на сервис приходится 50—70 % четырёхмиллиардного (в долларах США) рынка онлайн-дистрибуции игр. Статистика финансов корпорации «Valve» хранится в тайне, но по словам основателя — Гейба — компания «чрезвычайно прибыльна». В октябре 2011 года пресс-служба корпорации объявила о рекордной для «Valve» прибыли, на 200% превысившей прибыль предыдущего года. По данным «Forbes», «Valve» оценивается различными источниками в промежутке от 2 до 4 млрд долларов США. Майкл Пэчтер в 2011 году оценил корпорацию в 2,5 млрд долларов США. Михаил Хомич на страницах «Коммерсанта» утверждал, что чистая прибыль компании за 2013 год составила порядка 1 млрд долларов США. Индустрия компьютерных игр оказала позитивное влияние на экономику США и принесла 8 миллиардов долларов прибыли в 2004 году от продаж. Прибыль от продажи консольных игр в 2005 году достигла 12,6 миллиардов долларов США, и к 2010 году возрасла до 17,2 миллиардов. На данный момент мировой игровой рынок составляет 50 млрд долларов. На показатели влияют не только видеоигры для консолей и компьютеров. Сильно закрепились на этом рынке платформы на операционных системах «IOS» и «Android». На рисунке 2 показан рост и перспективы доли рынка мобильных игр и игр на ПК, консоли и другие платформы. Таким образом, обратим внимание, что 1-ое место занимает США-20,5 млрд \$, 2-ое место Китай-17,9 млрд \$, 3-е место Япония-12,2 млрд \$.

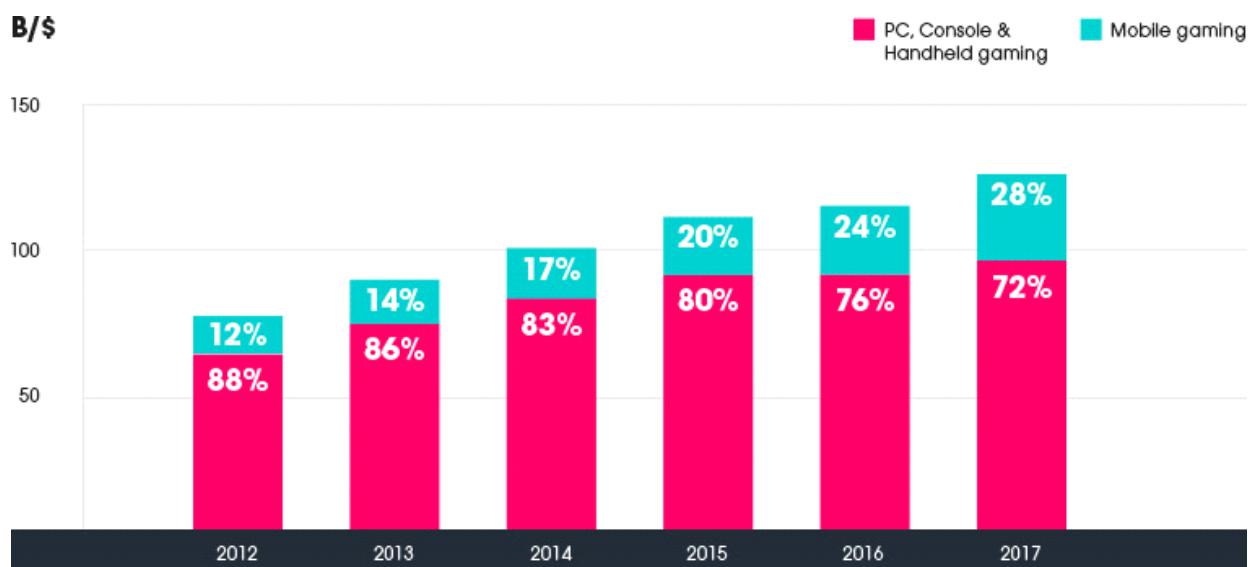


Рисунок 4.2 - Рост и перспективы доли рынка мобильных игр и игр на ПК, консоли и другие платформы

Таким образом, главные плюсы индустрии компьютерных развлечений в том, что продукция имеет высокую мобильность. Продукт с помощью множества сервисов можно приобрести из любой точки имеющей доступ к интернету. Так же индустрия слабо чувствительна к экономическим кризисам, она продолжает своё развитие даже во время кризисов. Выпускаемая продукция распространяется на большое количество разнообразных платформ, которыми пользуются большинство людей. Это расширяет и аудиторию, которая играет в игры. На данный момент возраст геймеров находится в диапазоне от 10 до 40 лет. И стоит сказать, что игровая индустрия несёт колоссальную прибыль странам в которых развита эта сфера. Уже сегодня ясно, что игры, сервисы, сети интернет, компьютеры и прочие технологические достижения в сфере компьютеризации прочно осели в нашей жизни, продолжают развиваться и стремительно интегрируется во все сферы нашей жизни включая экономику.

#### **4.2 Трудовые ресурсы, используемые в работе.**

В данной работе используется интеллектуальный труд, стоимость затрат которого выше, чем физического труда.

В разработке приняли участие:

- художник-дизайнер – художественная реализация проекта;
- программист - разработка алгоритмов и программирование;
- руководитель – оформление и проверка отчета.

Количество сотрудников задействованных в разработке представлено в таблице 4.1.

Таблица 4.1 – Сотрудники и их заработная плата

Должность	Количество человек	Заработная плата в месяц, тенге
Программист	1	120000
Художник-дизайнер	1	120000
Руководитель проекта	1	140000
Итого	3	380000

#### **4.3 Оборудование, используемое в работе**

Оборудование, используемое при разработке сайта представлено в таблице 4.2.



Таблица 4.2 – Перечень оборудования, необходимого для разработки

Наименование изделий	Характеристика	Количество единиц	Цена за единицу, тенге	Общая сумма, тенге
Ноутбук	Intel Core i7-5500U 6Gb DDR3 GeForce GT 940M 2GB HDD 1000Gb 15.6" LED LCD (16:9)	2	150000	300000
Многофункциональное устройство	Samsung SL-M2070	1	30000	30000
Итого				330000

Цены на оборудование представлены без учета НДС.

#### 4.4 Программное обеспечение, используемое в работе

При разработке интернет-магазина было использовано следующее программное обеспечение:

- Windows 10 – операционная система;
- Visual Studio 2015 Community – среда разработки;
- Adobe Photoshop CC – графический редактор;

Программное обеспечение, использованное при разработке сайта, представлено в таблице 4.3.

Таблица 4.3 – перечень программного обеспечения, необходимого для разработки интернет-магазина светотехники

Программное обеспечение	Стоимость, тенге
Windows 10	бесплатно (вместе с ноутбуками)
Adobe Photoshop CC	7000
Visual Studio 2015 Community	бесплатно
Итого	7000

Цены на ПО приведены без учета НДС.

## 4.5 Сроки реализации проекта

Процесс разработки и сроки разработки мобильной игры состоит из 6 этапов и включает в себя:

- создание дизайн-документа;
- создание прототипа;
- обновление дизайн-документа;
- производство;
- тестирование;
- формирование отчетов и поддержка.

Этапы и сроки реализации проекта отображены в таблице 4.4.

Таблица 4.4 - Этапы и сроки реализации проекта

Перечень работ		Недели от начала работ									
		1	2	3	4	5	6	7	8	9	10
1 этап	Постановка задачи	■									
	Формирование идеи	■	■								
	Описание сеттинга игры		■	■							
	Описание Core- механики игры		■	■							
2 этап	Разработка базового движка игры			■	■						
	Описание поведения физики игры				■	■					
3 этап	Обновление дизайн- документа с итогами прототипа					■	■	■			
4 этап	Анализ архитектуры приложения					■	■	■			
	Деальное кодирования мира игры						■	■	■		
	Отрисовка ключевых персонажей и задников							■	■	■	
5 этап	Тестирование ПО									■	
	Отладка ПО									■	

6	Оформление НИР											
этап	Проверка и сдача отчета											

#### 4.6 Расчет стоимости работы по разработке

Расчет стоимости работы по разработке - это наиболее важная часть экономического анализа, так как на основе этого расчета определяются затраты рабочего времени на разработку проекта на каждом этапе, а также трудовые затраты.

Затраты на разработку данного проекта определяются по формуле

$$C = \text{ФОТ} + O_c + A + \text{Э} + C_{\text{пр}} + H \quad (4.1)$$

где ФОТ - фонд оплаты труда;

$O_c$  - социальный налог;

A - амортизационные отчисления;

Э - затраты на электроэнергию;

$C_{\text{пр}}$  - прочие расходы;

H - накладные расходы.

##### 4.6.1 Расчет затрат на оплату труда

Затраты на оплату труда персонала, задействованного в разработке проекта, рассчитываются по формуле:

$$\text{ФОТ} = Z_{\text{осн}} + Z_{\text{доп}} \quad (4.2)$$

где  $Z_{\text{осн}}$  - основная заработная плата;

$Z_{\text{доп}}$  - дополнительная заработная плата.

Труд программиста, дизайнера и руководителя принят условно, на договорной основе в размере 120000, 120000 и 140000 тенге соответственно.

На этапах разработки программного продукта участники разработки сайта задействованы неравноценно, для этого необходимо рассчитать средний дневной заработок, а затем и общий размер заработной платы, в зависимости от их фактического участия.

Средний дневной заработок каждого работника рассчитывается по формуле:

$$D = \frac{ЗП_m}{D_p} \quad (4.3)$$

где  $ЗП_m$  – ежемесячный размер заработной платы;

$D_p$  – количество рабочих дней в месяце (это 26 дней – шестидневная рабочая неделя).

1) Для программиста

$$D = \frac{120000}{26} = 4615 \text{ тенге/день}$$

2) Для художника-дизайнера

$$D = \frac{120000}{26} = 4615 \text{ тенге/день}$$

3) Для руководителя проекта

$$D = \frac{140000}{26} = 5384 \text{ тенге}$$

Заработная плата за один час вычисляется по формуле:

$$H = \frac{ЗП_m}{D_p * Ч_p} \quad (4.4)$$

где  $ЗП_m$  – ежемесячный размер заработной платы;

$D_p$  – количество рабочих дней в месяце.

$Ч_p$  – Количество часов рабочего дня ( $Ч_p = 8$ )

1) Для программиста

$$H = \frac{120000}{26 * 8} = 577 \text{ тенге/час}$$

2) Для художника-дизайнера

$$H = \frac{120000}{26 * 8} = 577 \text{ тенге/час}$$

3) Для руководителя

$$H = \frac{140000}{26 * 8} = 673 \text{ тенге/час}$$

Длительность цикла в днях по каждому виду работ определяется по формуле:

$$t_n = \frac{T}{q_n * z * K} \quad (4.5)$$

где T – трудоемкость этапа, норма-час;

$q_n$  – количество исполнителей по этапу;

z – продолжительность рабочего дня, z = 8 часов;

K – коэффициент выполнения норм времени, K = 1,1.

Полученную величину  $t_n$  округляем в большую сторону до целых дней.

$$t_1 = \frac{24}{1 * 8 * 1,1} \approx 3 \text{ - программист, постановка задачи;}$$

Для каждого вида работ произведены аналогичные расчеты.

$$t_n = 3 + 4 + 3 + 4 + 6 + 3 + 4 + 8 + 13 + 6 + 2 + 2 + 1 + 1 = 60 \text{ дней}$$

Таким образом, для проведения всех работ необходимо 60 дней.

Сводные данные по расчету заработной платы персонала, задействованного в разработке проекта приведены в таблице 4.5.

Таблица 4.5 – Сводные данные по расчету основной заработной платы персонала задействованного в разработке проекта.

Наименование этапов	Исполнитель	Трудоемкость, норма-час	Длительность цикла, дни	Заработная плата за час работы, тенге	Сумма заработной платы, тенге
Постановка задачи	Программист	24	3	577	13848

Формирование идеи	Художник-дизайнер	32	4	577	18464
Описание сеттинга игры	Программист	24	3	577	13848
Описание Core-механики игры	Программист	32	4	577	18464
Разработка базового движка игры	Программист	48	6	577	27696
Описание поведения физики игры	Программист	24	3	577	13848
Обновление дизайн-документа с итогами прототипа	Художник-дизайнер	32	4	577	18464
Анализ архитектуры приложения	Программист	64	8	577	36928
Реальное кодирование мира игры	Программист	104	13	577	60008
Отрисовка ключевых персонажей и задников	Художник-дизайнер	48	6	577	27696
Тестирование ПО	Программист	16	2	577	9232

Отладка ПО	Программист	16	2	577	9232
Оформление НИР	Руководитель	8	1	673	5384
Проверка и сдача отчета	Руководитель	8	1	673	5384
Итого		480	60	8270	278496

Дополнительная заработная плата составляет 10% от основной заработной платы и вычисляется по формуле:

$$Z_{\text{доп}} = Z_{\text{осн}} * 0,1 \quad (4.6)$$

и составит

$$Z_{\text{доп}} = 278496 * 0,1 = 27849 \text{ тенге};$$

Таким образом, затраты на оплату труда согласно произведенным расчетам и в соответствии с формулой 4.2 составит

$$\text{ФОТ} = 278496 + 278496 = 306345 \text{ тенге}$$

#### 4.6.2 Расчет затрат по социальному налогу

Социальный налог составляет 11% (ст. 358 п. 1 НК РК) от дохода работника, и рассчитывается по формуле

$$O_c = (\text{ФОТ} - \text{ПО}) * 11\% \quad (4.7)$$

где ПО – пенсионные отчисления, которые составляют 10% от ФОТ и социальным налогом не облагаются, вычисляются отчисления по формуле

$$\text{ПО} = \text{ФОТ} * 10\% \quad (4.8)$$

$$\text{ПО} = 306345 * 0,1 = 30634 \text{ тенге.}$$

Таким образом, в соответствии с произведенными расчетами и согласно формуле 4.7 размер отчислений на социальные нужды составит

$$O_c = (306345 - 30634) * 0,11 = 30328 \text{ тенге}$$

#### 4.6.3 Расчет амортизационных отчислений

Амортизационные отчисления рассчитываются по формуле

$$A_i = \frac{H_A * C_{\text{ПЕР}} * N}{100 * n} \quad (4.9)$$

где  $H_A$  – норма амортизации;

$C_{ПЕР}$  – первоначальная стоимость оборудования;

$N$  – количество дней на выполнение работ;

$n$  – количество рабочих дней в году.

Норма амортизации на компьютерную технику составляет 40% от стоимости всего оборудования, на программное обеспечение - 15%.

Таким образом, амортизационный отчисления по используемому оборудованию, в соответствии с формулой 4.9 составят

- на ноутбук

$$A_1 = \frac{40 * 150000 * 2 * 60}{100 * 365} = 19726 \text{ тенге}$$

- на многофункциональное устройство

$$A_2 = \frac{40 * 30000 * 4}{100 * 365} = 131 \text{ тенге}$$

$$A_i = 19726 + 131 = 19857 \text{ тенге}$$

Сводные результаты расчета амортизационных отчислений представлены в таблице 4.6.

Таблица 4.6 – Сводные данные по расчету затрат на амортизацию

Наименование оборудования	Количество	Норма амортизации, %	Сумма амортизации, тенге
Ноутбук	2	40	19726
Многофункциональное устройство	1	40	131
Итого			19857

#### 4.6.4 Расчет затрат на электроэнергию

Поскольку в процессе производства используется электрооборудование, необходимо рассчитать затраты на электроэнергию. Затраты на электроэнергию для производственных нужд включают в себя расходы электроэнергии на оборудование и дополнительные нужды. И рассчитываются по формуле



$$\mathcal{E} = \mathcal{Z}_{\text{эл.эн.об.}} + \mathcal{Z}_{\text{доп.}} \quad (4.10)$$

где  $\mathcal{Z}_{\text{эл.эн.об.}}$  – затраты на электроэнергию для оборудования;  
 $\mathcal{Z}_{\text{доп.}}$  – затраты на электроэнергию для дополнительных нужд.

Расходы электроэнергии для оборудования рассчитываются по формуле

$$\mathcal{Z}_{\text{эл.эн.об.}} = W * T * S * K_{\text{исп}} \quad (4.11)$$

где  $W$  – потребляемая мощность, Вт;

$T$  – время работы, часы;

$S$  – тариф (1кВт = 16,02 тенге);

$K_{\text{исп}}$  – коэффициент использования ( $K_{\text{исп}} = 0,9$ ).

$$\mathcal{Z}_{\text{эл.эн.об. (ноутбук)}} = 0,7 * 480 * 16,02 * 0,9 * 2 = 9688,9 \text{ тенге}$$

$$\mathcal{Z}_{\text{эл.эн.об. (МФУ)}} = 0,9 * 32 * 16,02 * 0,9 = 415,24 \text{ тенге}$$

Общая сумма затрат на электроэнергию основного оборудования согласно формуле 4.11 составляет

$$\mathcal{Z}_{\text{эл.эн.об.}} = 9688,9 + 415,24 = 10104,14 \text{ тенге}$$

Затраты на дополнительные нужды берутся по показателю от затрат на оборудование в размере 5% и рассчитывается по формуле

$$\mathcal{Z}_{\text{доп.}} = \mathcal{Z}_{\text{эл.эн.об.}} * 5\% \quad (4.12)$$

и составляют

$$\mathcal{Z}_{\text{доп.}} = 10104,14 * 0,05 = 505,2 \text{ тенге}$$

Таким образом, суммарные затраты на электроэнергию, согласно формуле 4.11 составляют

$$\mathcal{E} = 10104,14 + 505,2 = 10609,34 \text{ тенге}$$

Сводные результаты расчета затрат на электроэнергию представлены в таблице 4.7

Таблица 4.7 – Сводные данные о затратах на электроэнергию

Наименование	Количество	Потребляемая	Число рабочих	Коэффициент	Время работы	Сумма затрат,
--------------	------------	--------------	---------------	-------------	--------------	---------------

приборов		мощность, Вт	дней	использования	оборудования, часы	тенге
Ноутбук	2	0,7	60	0,9	480	10104,14
МФУ	1	0,9	4	0,9	32	505,2
Итого						10609,34

#### 4.6.5 Расчет накладных и прочих расходов

Прочие расходы включают в себя:

1) расходы на интернет на 60 дней(2месяца). Стоимость интернета за один месяц составляет 4500 тенге. Получаем:

$$4500 * 2 = 9000 \text{ тенге}$$

2) расходы на канцелярские товары:

- упаковка бумаги формата А4 – 1000 тенге;
- упаковка ручек – 300 тенге;
- 3 карандаша – 150 тенге;
- ластик – 50 тенге;
- линейка - 100 тенге;
- стикеры – 200 тенге.

Общая стоимость канцелярских товаров

$$1000 + 300 + 150 + 50 + 100 + 200 = 1800 \text{ тенге.}$$

3) аренда помещения на 60 дней(2месяца).

Площадь помещения:

$$S = 6\text{м} * 6\text{м} = 36 \text{ м}^2.$$

Стоимость одного квадратного метра равна 1500 тенге. Получаем:

$$36 * 1500 * 2 = 108000 \text{ тенге.}$$

Прочие расходы составляют

$$C_{\text{ПР}} = 9000 + 1800 + 108000 = 118800 \text{ тенге.}$$

Накладные расходы составляют 50% от всех затрат и рассчитываются по формуле

$$H = (\text{ФОТ} + O_c + A + \text{Э} + C_{\text{ПР}}) * 50\% \quad (4.13)$$

$$H = (306345 + 30328 + 19587 + 10609 + 118800 + 7000) * 0,5 =$$

$$= 246334 \text{ тенге}$$

#### 4.6.6 Расчет стоимости по всем статьям затрат и определение структуры затрат

В соответствии с формулой 4.1 суммарные затраты по разработке моего проекта составляют

$$C = 306345 + 30328 + 19587 + 10609 + 118800 + 7000 + 246334 =$$

$$= 739003 \text{ тенге}$$

Смета затрат по разработке мобильной игры, а также структура расходов представлены в таблице 4.8 и на рисунке 4.1

Таблица 4.8 – Суммарные данные по стоимости разработки проекта

Наименование статьи затрат	Сумма, тенге
Фонд оплаты труда	306345
Социальный налог	30328
Амортизация	19587
Затраты на электроэнергию	10609
Прочие расходы	125800
Накладные расходы	246334
Итого	739003

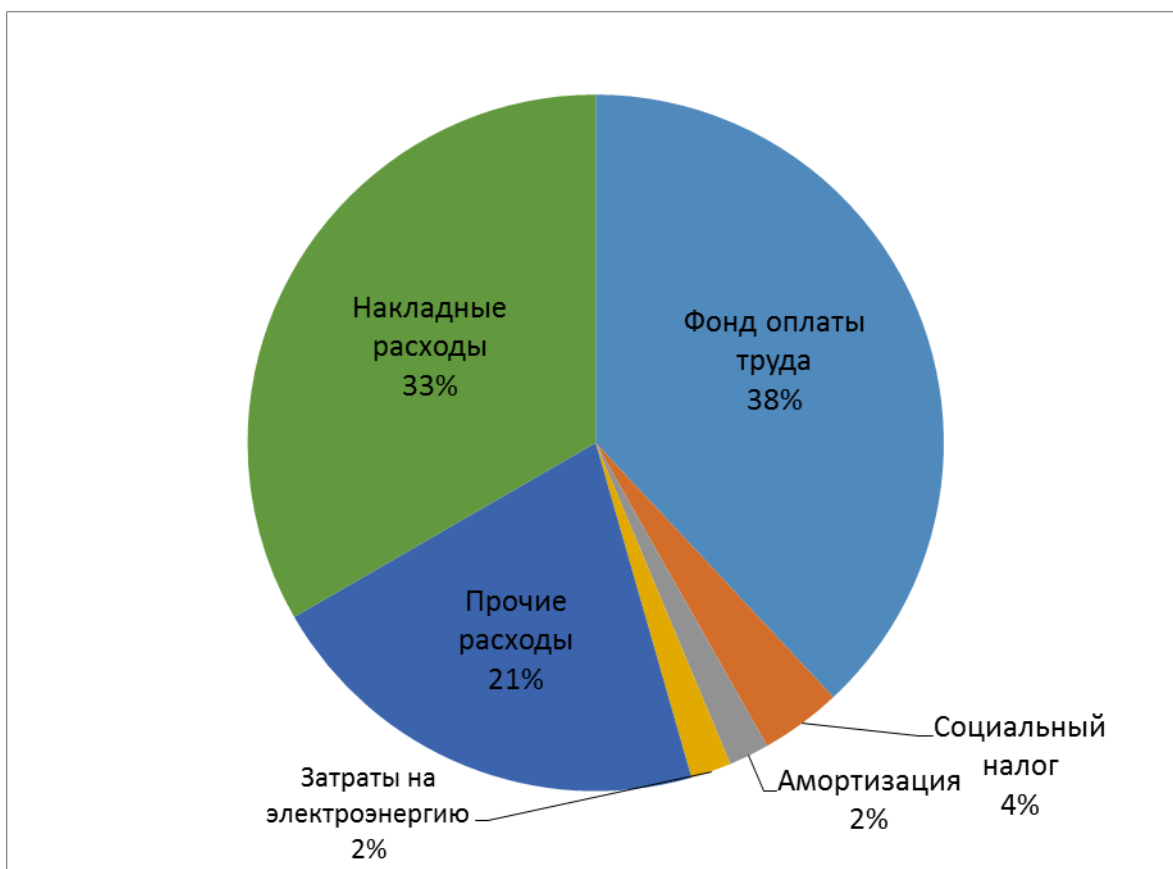


Рисунок 4.1 – Структура затрат по разработке мобильной игры

#### 4.7 Цена интеллектуального руда

Цена реализации проекта складывается из стоимости и чистого дохода

$$Ц = С + П \quad (4.14)$$

где С – стоимость продукта;

П – чистый доход.

При определении первоначальной цены следует задаться желаемым уровнем рентабельности (здесь 20%) реализации программных продуктов

$$Ц_{П} = С * \left(1 + \frac{P}{100}\right) \quad (4.15)$$

где P – рентабельность.

$$Ц_{П} = 739003 * \left(1 + \frac{20}{100}\right) = 886803 \text{ тенге}$$

Цена реализации проекта рассчитывается по формуле

$$C_p = C_{\text{п}} + \text{НДС} \quad (4.16)$$

где НДС – налог на добавленную стоимость по ставке 12%

$$\text{НДС} = C_{\text{п}} * 12\% \quad (4.17)$$

$$\text{НДС} = 886803 * 0,12 = 106416 \text{ тенге}$$

В соответствии с формулой 4.16 цена реализации проекта составит

$$C_p = 886803 + 106416 = 993219 \text{ тенге}$$

В зависимости от спроса, окончательная цена на интеллектуальный труд в условиях рыночных отношений будет варьироваться и изменяться.

#### **4.8 Вывод**

Разработка игр сложное дело и ничем не уступает разработке программного обеспечение. В проекте требуется специалисты разных направлений и слаженный процесс работы.

Рынок игровой индустрии испытывает неуклонный рост, а тем временем местный рынок почти никем не занят. Расчитав целесообразность проекта, можно прийти к выводу, что проект является успешным, после которого можно будет реализовать чуть более масштабный проект, тем самым повышая уровень качества.

## **6 Безопасность жизнедеятельности.**

### **6.1 Анализ потенциально опасных вредных факторов в помещении при разработке**

Главной целью данного проекта является компьютерной и мобильной игры, с целью развития местной игровой индустрии.

Работа с компьютером характеризуется значительным умственным напряжением и нервно-эмоциональной нагрузкой операторов, высокой напряженностью зрительной работы. Большое значение имеет рациональная конструкция и расположение элементов рабочего места, что важно для поддержания оптимальной рабочей обстановки сотрудника.

В процессе работы с компьютером необходимо соблюдать правильный режим труда и отдыха. В противном случае у персонала отмечаются значительное напряжение зрительного аппарата с появлением жалоб на неудовлетворенность работой, головные боли, раздражительность, нарушение сна, усталость и болезненные ощущения в глазах, в пояснице, в области шеи и руках.

Оборудование малошумящее – вредность в качестве повышенного шума отсутствует. Повышенный уровень электромагнитных излучений отсутствует. Здание относится к I степени огнестойкости (СНиП РК 2.02-05-2002) (Здания с несущими и ограждающими конструкциями из естественных или искусственных материалов, бетона или железобетона с применением листовых негорючих материалов). Рабочее помещение по вопросам пожарной безопасности относится к классу “Д”. В соответствии с типовыми правилами пожарной безопасности административные здания и отдельные помещения, и технологические установки обеспечиваются первичными средствами пожаротушения согласно нормативам.

Проведем анализ освещенности. Правильно спроектированное и выполненное производственное освещение улучшает условия зрительной работы, снижает утомляемость, способствует повышению производительности труда, благотворно влияет на производственную среду, оказывая положительное психологическое воздействие на работающего, повышает безопасность труда и снижает травматизм.

Недостаточность освещения приводит к напряжению зрения, ослабляет внимание, приводит к наступлению преждевременной утомленности. Чрезмерно яркое освещение вызывает ослепление, раздражение и резь в глазах. Неправильное направление света на рабочем месте может создавать резкие тени, блики, дезориентировать работающего. Все эти причины приводят к

несчастному случаю или профзаболеваниям, поэтому столь важен правильный расчет освещенности.

В помещении расположены два окна, и естественное освещение соответствует нормам. Естественное освещение не обеспечивает в течение всего рабочего времени необходимого освещения, особенно в зимнее время (режим работы: с 9:00 до 18:00), поэтому в рабочем помещении предусмотрена система искусственного общего освещения, состоящая из светильников с люминесцентными лампами.

Проведем анализ системы вентиляции. Компьютеры, установленные в рабочем помещении не является источником выделения тепла (очень незначительное выделение тепла аппаратурой никаким образом не оказывает влияние на микроклимат рабочего помещения).

Климатические условия эксплуатации оборудования полностью совпадают климатическими условиями, нормируемыми для рабочего персонала.

Поскольку климат рабочего помещения не соответствует принятым нормативам, то необходимо для обеспечения нормальных условий микроклимата в помещении оборудовать его дополнительно системой кондиционирования.

Перспективным с точки зрения создания нормальных микроклиматических условий в рабочей зоне является использование кондиционирующих установок.

План рабочего помещения представлен на рисунке 1.1.

Помещение имеет следующие параметры:

находится на первом этаже пятиэтажного здания;

размеры помещения (комнаты): длина 5 м, ширина 6 м, высота 3м;

вид светопропускающего материала – стекло листовое, двойное;

дверь – деревянная открывающаяся;

солнцезащитные устройства – убирающиеся регулируемые жалюзи и шторы;

окно размером 1,5\*2;

помещение имеет 1 кондиционер Samsung;

внутренняя отделка стен – светлая;

помещение по зрительным условиям работы относится к категории легких работ (легкая физическая, категория 1а);

искусственное освещение – 4 люминесцентных ламп.

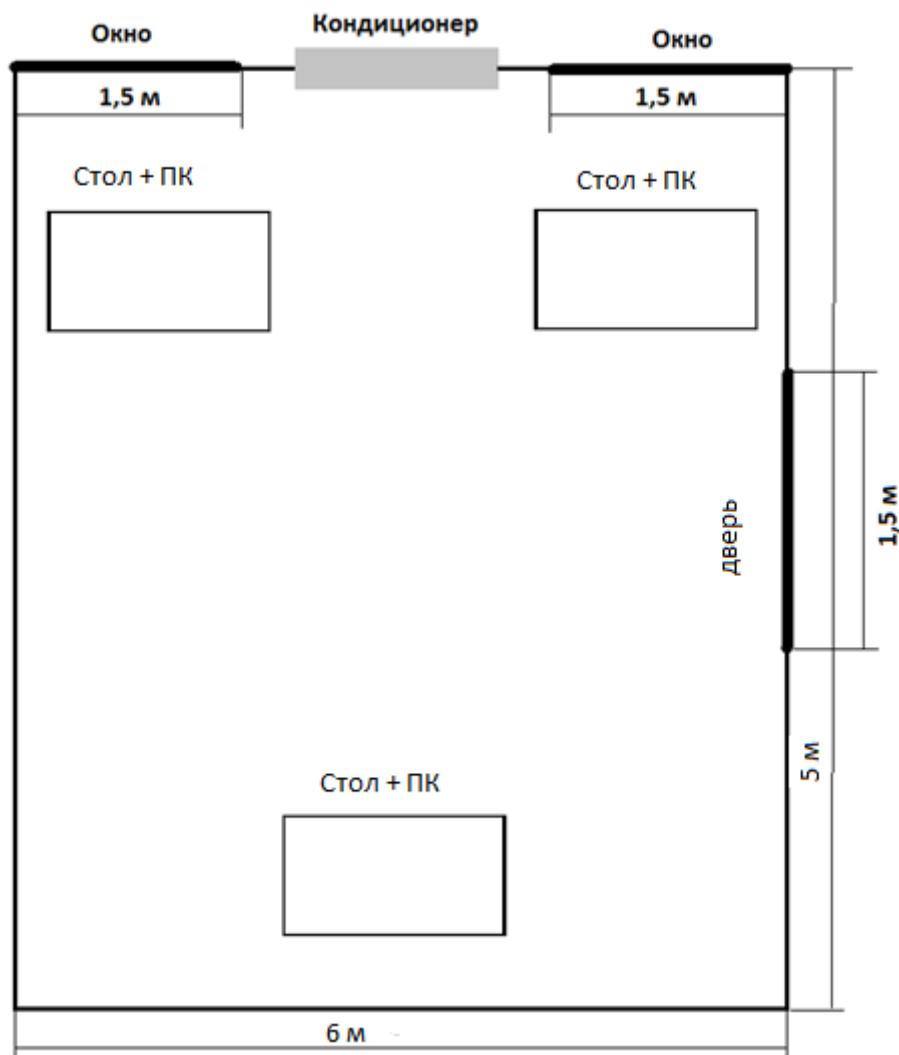


Рисунок 6.1 – План помещения

К организации рабочих мест предъявляются особые требования:

Рабочее место на одного человека должно быть не менее 6 м<sup>2</sup>, а объем - не менее 24 м<sup>3</sup>.

Размещать компьютеры экраном к окну не допускается, так как это создаёт тень и блики на экране монитора.

На основе полученных данных можно квалифицировать вид выполняемых работ.



Таблица 6.1 – Виды работ (ГОСТ 12.2.032-78)

Наименование работ	Класс работ	Пол работника	Высота рабочей поверхности	Высота сиденья
Легкие работы (конторская работа)	Класс Ia (работа, выполняемая в сидячем положении)	Мужской, женский	725 мм	420 мм

Проведем оценку рабочего места:

выполняемая работа относится к категории легких физических, категория Ia, работа производится сидя и не требует физического напряжения (ГОСТ 12.2.032-78);

рабочее место составляет 30 м<sup>2</sup> и объем – 90 м<sup>3</sup>;

число работников – 3.

## 6.2 Расчет системы вентиляции

Расчет системы вентиляции ведем согласно ГОСТ 12.1.005 – 88 ССБТ «Воздух рабочей зоны. Общие требования» и Хакимжанову Т. Е. «Безопасность жизнедеятельности. Расчет аспирационных систем. Методические указания к выполнению раздела в дипломных проектах».

Для вентиляции рабочего помещения используются каналы естественной вентиляции, прокладываемые при строительстве здания и открытые окна летом. Однако такая вентиляция не позволяет поддерживать климатические параметры рабочего помещения в пределах нормы (таблица 1.2) в условиях климата города Шардара (в особенности – летом).

Таблица 6.2 – ГОСТ 12.1.005 – 88 ССБТ «Воздух рабочей зоны. Общие требования»

Период года	Температура воздуха, °С	Влажность воздуха, %, не более	Скорость движения воздуха, не более м/с
Теплый	20-22	60-40	0,1
Холодный	20-22	45-30	0,1

Определим необходимое количество кондиционеров для создания комфортных условий труда в помещении. Расчет произведем по методике для теплого времени года. В помещении за счет тепловыделений производственного оборудования могут иметь место значительные избытки тепла (разность между

тепловыделениями в помещении и теплоотдачей через стены, окна, двери и т.д.), удаление которых, прежде всего, должна обеспечить система вентиляции.

Рассчитаем в нашем помещении со значительными тепловыделениями количество приточного воздуха, м<sup>3</sup>/ч, необходимого для поглощения избытка тепла.

Сначала определим тепловыделения от аппаратуры:

$$Q_1 = \eta_1 \cdot \eta_2 \cdot \eta_3 \cdot \eta_4 \cdot N_{\text{НОМ}} \quad (6.1)$$

$$Q_1 = 0,8 \cdot 0,6 \cdot 0,6 \cdot 0,75 \cdot 1200 = 259 \text{ кДж/ч}$$

где  $\eta_1 = 0,8$  – коэффициент использования номинальной мощности;

$\eta_2 = 0,6$  – коэффициент загрузки, т.е. отношение средней потребляемой мощности к максимально необходимой;

$\eta_3 = 0,6$  – коэффициент одновременной работы аппаратуры;

$\eta_4 = 0,75$  – коэффициент, характеризующий долю механической энергии, превратившейся в тепло;

$N_{\text{НОМ}} = 1200 \text{ кДж/ч}$  – номинальная мощность всей аппаратуры.

Тепловыделение от источников искусственного освещения рассчитывают, предполагая, что практически вся затрачиваемая энергия, в конечном счете, преобразуется в тепло, по формуле:

$$Q_2 = \varphi \cdot N_{\text{осв}} \quad (6.2)$$

$$Q_2 = 0,8 \cdot 4 \cdot 60 = 192 \text{ Дж/ч}$$

где  $\varphi = 0,8$  – коэффициент, учитывающий количество энергии переходящей в тепло;

$N_{\text{осв}} = 4 \cdot 60 = 300 \text{ Вт}$  – мощность осветительной установки помещения (4 лампы по 60 Вт каждая).

Тепловыделение от людей определяют по формуле:

$$Q_3 = n \cdot q_n \quad (6.3)$$

$$Q_3 = 3 \cdot 120 = 360 \text{ Дж/ч}$$

где  $n=3$  – число работников;

$q_n = 120 \text{ Вт}$  – количества тепла, выделяемого одним человеком.

Количества тепла, поступающего в помещение через окна от солнечной радиации, определяют по формуле:

$$Q_4 = F_{\text{ост}} \cdot q \cdot m \cdot k \quad (6.4)$$

$$Q_4 = 3 \cdot 224 \cdot 2 \cdot 1,25 = 1680 \text{ Дж/ч}$$

где  $F_{\text{ост}} = 1,5 \cdot 2 = 3 \text{ м}^2$  – площадь окна;

$q=224 \text{ Вт/м}^2$  – теплопоступление через  $1 \text{ м}^2$  окна (окна выходят на юг);

$m=2$  – число окон;

$k=1,25$  – поправочный множитель для металлического переплета;

Избыточное тепловыделение определяется по формуле

$$Q_{\text{изб}} = Q_1 + Q_2 + Q_3 + Q_4 \quad (6.5)$$

$$Q_{\text{изб}} = 540 + 192 + 360 + 1680 = 2532 \text{ Дж/ч}$$

где  $Q_1$  – тепловыделение от аппаратуры;

$Q_2$  – тепловыделение от источников освещения;

$Q_3$  – тепловыделение от людей;

$Q_4$  – теплопоступление от солнечной радиации сквозь окна.

Определим количество приточного воздуха, необходимого для поглощения избытка тепла:

$$L = \frac{\Sigma Q_{\text{изб}}}{c \cdot (t_{\text{уд}} - t_{\text{пр}}) \cdot \gamma_{\text{пр}}} \quad (6.6)$$

$$L = \frac{2532}{1 \cdot (27 - 20) \cdot 1,2} = 301,4 \text{ м}^3/\text{ч}$$

где  $Q_{\text{изб}} = 2532 \text{ Дж/ч}$  – избыточное выделение явной теплоты;

$C=1 \text{ Дж/(кг*град)}$  – теплоемкость воздуха;

$t_{\text{уд}} = 27 \text{ }^\circ\text{C}$  – температура удаляемого из помещения воздуха;

$t_{\text{пр}} = 20 \text{ }^\circ\text{C}$  – температура приточного воздуха;

$\gamma_{\text{пр}} = 1,2 \text{ кг/м}^3$  – плотность приточного воздуха,  $1,2 \text{ кг/ м}^3$ .

В соответствии с полученными расчетами выбираем кондиционер. Для выполнения этих целей и обеспечения притока воздуха применим настенный кондиционер Samsung MH18ZC2.

Серия кондиционера Samsung MN18ZC2, предназначенная для помещений площадью до 80 м<sup>2</sup>. Эти кондиционеры отличаются повышенной эффективностью охлаждения и обогрева, быстро обеспечивая заданные температурные параметры воздуха, удаляет из воздуха наиболее вредные соединения – аммиак, формальдегид, сероводород. Активный угольный фильтр задерживает мельчайшие частички пыли и удаляет неприятные запахи. Технические характеристики кондиционера приведены в таблице 6.3.

Таблица 6.3 – Характеристики кондиционера

Площадь не более, кв. м	60
Пульт д/у	есть
Мощность охлаждения/нагрев, кВт	5,26/5,56
Питание (В/Гц)	220/50
Потребляемая мощность охлаждение/нагрев, кВт	1,78
Рабочий ток холод/тепло, А	9
Уровень шума, дБ	37
Расход воздуха не более, м <sup>3</sup> /мин (м <sup>3</sup> /час)	10 (600)

При расчете системы вентиляции полностью выполнены требования ГОСТ 12.1.005 – 88 ССБТ «Воздух рабочей зоны. Общие требования». В системе вентиляции применен кондиционер, обеспечивающий комфортные условия, т.е. создающий и автоматически поддерживающий температуру, влажность, чистоту и скорость движения воздуха отвечающий оптимальным санитарно-гигиеническим требованием для работы в помещении.

## **6.3 Расчет системы освещения**

### **6.3.1 Расчет естественного освещения**

Естественное освещение не обеспечивает в течение всего рабочего времени необходимого освещения, так как может измениться погода, либо работы могут быть в позднее время, когда уже темнеет и естественного освещения может быть не достаточно, поэтому в рабочем помещении предусмотрена система искусственного общего освещения, состоящая из светильников с люминесцентными лампами. Нормативы на источники света приведены в таблице 6.4.

Таблица 6.4 – Рекомендуемые источники света при системе общего освещения (СНиП РК 2.04.-05-2002)

Характеристика зрительной работы по требованию к цветоразличию	Освещенность, лк	Диапазон цветов температуры источника света $T_c, ^\circ K$	Применяемый тип источника света
Различие цветных объектов при невысоких требованиях к цветоразличию	300, 400	3500 - 5500	ЛД, ЛДЦ, ЛБ,
	150, 200	3000 - 4500	ЛБ, (ЛХБ), НЛВД+МТЛ, ДРЛ

Освещённость, необходимая для нормального выполнения работ в данном помещении: 400 лк. Для этого в помещении используются 4 люминесцентные лампы белого цвета. Этот выбор обусловлен тем, что люминесцентные лампы более экономичны, чем обычные лампы накаливания. Известно, что различный спектральный состав по-разному влияет на настроение человека. Лампы “теплого” цвета способствуют расслаблению, их лучше использовать в спальнях, местах для отдыха. Лампы “холодного” света способствуют зрительному и психоэмоциональному комфорту, их лучше использовать в гостиных, местах приема гостей и кухнях. Лампы “дневного” света способствуют концентрации внимания и увеличения работоспособности, их лучше использовать в кабинетах, местах, предназначенных для умственной работы. Так как лампы будут использоваться в рабочем помещении, были выбраны люминесцентные лампы белого цвета ЛД-60.

Рассчитаем площадь боковых световых проемов помещения, необходимую для создания нормируемой освещенности на рабочем месте. Помещение имеет размеры: длина  $L = 6\text{ м}$ , ширина  $B = 5\text{ м}$ , высота  $H = 3\text{ м}$ . Высота рабочей поверхности над уровнем пола  $h_{рп}$ ,  $h_{рп} = 0,725\text{ м}$ , окна начинаются с высоты  $h_{но}$ ,  $h_{но} = 0,8\text{ м}$ , высота окна  $h_o$ ,  $h_o = 2\text{ м}$ . Рабочее помещение находится в

IV часовом поясе – в г. Алматы (пояс светового климата - IV 50<sup>0</sup> северной широты и южнее (Алматы, Караганда)).

Рабочее место расположено в  $l_{рт}, l_{рт} = 1,5 м$  от наружной стены помещения, где проектируем оконные проемы. Минимальная освещенность будет в точке, отстоящей на расстояние 4 м от оконного проема.

Общую площадь окон  $S_0, м^2$ , определим по формуле:

$$100 \cdot \frac{S_0}{S_n} = \frac{e_n \cdot \eta_0}{\tau_0 \cdot r_1} \cdot k_{зд} \cdot k_3 \quad (6.7)$$

$$S_0 = \frac{S_n \cdot e_n \cdot \eta_0}{100 \cdot \tau_0 \cdot r_1} \cdot k_{зд} \cdot k_3 \quad (6.8)$$

где:  $S_n$  – площадь помещения  $м^2$ ;

$S_n = 30 м^2$ ;

$e_n$  – нормированное значение КЕО, выбираемое по таблице 6.5.

Таблица 6.5 – Нормируемые показатели естественного, искусственного и совмещенного освещения помещений жилых зданий (СНиП РК 2.04.-05-2002)

Помещения	Естественное освещение при боковом освещении КЕО $e_n, \%$
Административные здания: Кабинеты, рабочие комнаты, офисы, представительства	1,2

Для высокой точности зрительных работ принимаем  $e_n = 1,2$

$m_N$  - коэффициент светового климата, который находится по таблице 6.6.

Таблица 6.6 – Значение коэффициента климата

Ориентация световых проемов по сторонам горизонта	Коэффициент светового климата, $m$
С	0,9

Учитывая заданный световой пояс, приняв ориентацию световых проемов на Север, определим:  $m_N=0,9$ .

$$e_H^{IV} = e_H \cdot m \cdot c \quad (6.9)$$

$c = 0,75$  (в наружных стенах зданий);

$$e_H^{IV} = 1,2 \cdot 0,9 \cdot 0,75 = 0,78 ;$$

Учитывая тип помещения, коэффициент запаса будет равен  $k_3 = 1,4$  (кабинеты и рабочие помещения, лаборатории, конструкторские бюро);

$\tau_0$  - общий коэффициент светопропускания равный

$$\tau_0 = \tau_1 \cdot \tau_2 \cdot \tau_3 \cdot \tau_4 \quad (6.10)$$

$\tau_1 = 0,5$  (пустотелые стеклянные блоки);

$\tau_2 = 0,6$  (деревянные двойные раздельные переплеты);

$\tau_3 = 0,8$  (железобетонные фермы и арки);

$\tau_4 = 1$  (убирающиеся регулируемые жалюзи и шторы);

$$\tau_0 = 0,5 \cdot 0,6 \cdot 0,8 \cdot 1 = 0,24 ;$$

$\eta_0$  - световая характеристика окон.

Определяем  $\eta_0$ :

$$l = B - 1$$

$$l = 5 - 1 = 4 \text{ м}$$

$$\frac{L}{l} = \frac{L}{B-1} = \frac{6}{4} = 1,5$$

$$h_{\text{расч}} = h_{\text{но}} + h_o - h_{\text{рп}} \quad (6.11)$$

$$h_{\text{расч}} = 0,8 + 2 - 0,72 = 2,08$$

$$\frac{B}{h_{\text{расч}}} = \frac{5}{2,03} = 2,5$$

По отношению ширины к глубине помещения получим, что  $\eta_0 = 10$ .

$\tau_1$  – коэффициент, учитывающий повышение КЕО при боковом освещении благодаря свету, отраженному от поверхностей помещения и подстилающего слоя, прилегающего к зданию.

Средний коэффициент отражения в помещении  $\rho_{\text{ср}} = 0,5$ , принимаем одностороннее боковое освещение.

$$\frac{\rho_{\text{ст}}}{B} = \frac{0,5}{5} = 0,1$$

Тогда  $\tau_1 = 1,05$

$k_{\text{зд}}$  – коэффициент, учитывающий затенение окон противостоящими зданиями.

Поскольку затеняющих зданий поблизости нет, то  $k_{\text{зд}} = 1$ .

Вычислим общую площадь окон:

$$S_0 = \frac{30 \cdot 1,4 \cdot 10 \cdot 1 \cdot 0,78}{100 \cdot 0,24 \cdot 1,05} = 13 \text{ м}^2$$

Площадь световых проемов равна  $S_{\text{сп}} = 13 \text{ м}^2$

Таким образом, площадь световых проемов ( $1,5 \cdot 2 \cdot 2 = 6 < 13$ ) не обеспечивает необходимых условий труда на рабочем месте. С целью создания наиболее благоприятных условий труда в помещении с параметрами  $5 \times 6 \times 3$  для обеспечения нормированного значения КЕО,  $e_N = 0,84$  при III характеристике зрительных работ, совместно с естественным освещением используется искусственное освещение.

### 6.3.2 Расчет искусственного освещения точечным методом

Расчет искусственного освещения методом коэффициента использования отдела разработки.

В процессе выполнения расчетной части необходимо выбрать систему освещения, источник света, тип светильника для заданного участка или рабочего помещения.

Следует определить количество светильников необходимых для обеспечения  $E_{\text{min}}$  и мощность осветительной установки, необходимых для обеспечения в рабочей аудитории нормированной освещенности.

Разряд зрительной работы - V. Нормируемая освещенность по таблице 6.7 - 400 лк.



Т а б л и ц а 6.7 - Технические характеристики люминесцентных ламп Т8

Номинальная мощность, Вт	Номинальный световой поток ламп типа Т8	Размер ламп, м	
		Диаметр	Длина штырькам по
36	3250	25,78	12

В качестве светильника возьмем **Т8 Luxline Plus F36W/865**. Длина светильника 1200 мм, ширина 140 мм.

Расчёт искусственного освещения производим методом коэффициента использования.

Коэффициенты отражения от потолка стен и пола соответственно равны:

$$\rho_{ПОГ} = 70\%;$$

$$\rho_{СТ} = 50\%;$$

$$\rho_{ПОЛ} = 30\%.$$

Вычислим высоту подвеса светильника над рабочей поверхностью

$$H = h - h_p - h_c \quad (6.12)$$

где  $h_c$ - расстояние от светильника до перекрытия,  $h_c = 0,11$  м;

$h_p$ - высота рабочей поверхности над полом;

$h_p = 0,8$  м.  $h$ -высота помещения,  $h = 3,2$  м.

$$H = 3,2 - 0,8 - 0,11 = 2,29\text{м}$$

Наивыгоднейшее расстояние от окна до светильника определяется как

$$L = \lambda \cdot H \quad (6.13)$$

где  $\lambda = 1,2 \div 1,4$

$$L = 1,2 \cdot 2,29 = 2,748\text{м}$$

Расстояние от стены до ближайшего светильника, когда работа у стены не проводится, определяем по формуле

$$l_1 = (0,4 \div 0,5) \cdot L \quad (5.14)$$

$$l_1 = 0,4 \cdot 2,748 = 1,1 \text{ м}$$

Определим индекс помещения

$$i = \frac{l \cdot s}{H(l+s)} \quad (6.15)$$

$$i = \frac{6 \cdot 4}{2,29(6+4)} = 1,048$$

Коэффициент использования в данном случае равен  $\eta = 49$ , коэффициент запаса равен  $k_3 = 1,2$ .

Определим количество люминесцентных ламп по формуле

$$N = \frac{E_H \cdot S \cdot Z \cdot K_3}{n \cdot \Phi_{л} \cdot \eta} \quad (6.16)$$

где  $S$ - площадь помещения;

$k_3$ - коэффициент запаса;

$E$ -заданная минимальная освещенность,  $E=400$  лк;

$Z$ -коэффициент неравномерности освещения,  $Z=1,1$ ;

$n$ -количество ламп в светильнике;

$\Phi_{л}$ - световой поток выбранной лампы,  $\Phi_{л}=3250$  лм;

$\eta$  - коэффициент использования,  $\eta = 49$ .

$$N = \frac{400 \cdot 24 \cdot 1,1 \cdot 1,2}{2 \cdot 3250 \cdot 0,49} \approx 4$$

Всего для создания нормируемой освещенности 400 лк необходимо 4 светильника по 2 люминесцентные лампы в каждом, мощность каждой лампы должна быть 36 Вт, значит, не требуется увеличивать или уменьшать количество имеющихся ламп, для соответствия санитарным нормам.

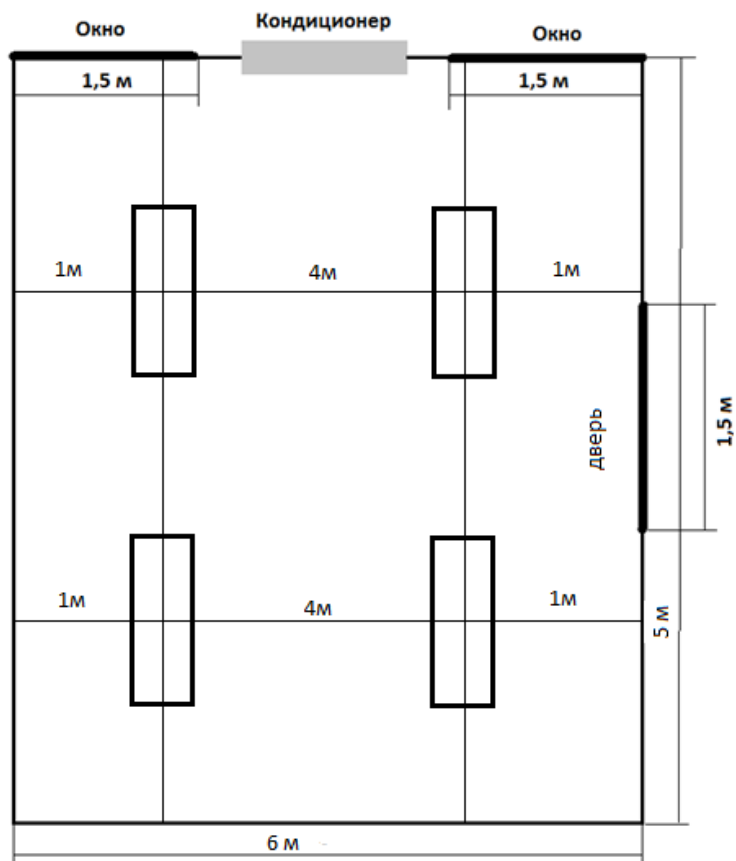


Рисунок 6.2 - План помещения светильников

**Заключение**

В данной работе я разработал кроссплатформенную игру. Усовершенствовал существующую игровую механику, также для работы игры были разработаны определенные правила. Была нарисована уникальная графика и сочинены игровые звуки.

Выбор технологий для проекта основывался на уже полученных знаниях в университете, а также учитывались потребности рынка. В качестве языка программирования использовался достаточно популярный и распространенный C#, а для работы с графикой открытый и широко применяемый в индустрии Monogame Framework.

Был проведен анализ игровой индустрии в экономической жизни государства. Выявлена связь экономической развитости государства с развитостью игровой индустрии. Проект был разработан с целью окупаемости и выхода на чистую прибыль.

Перед началом работы были выявлены опасные факторы при процессе производства продукта. Был разработан план по их предотвращению и снижению до низкого незначительного эффекта.

## **Список литературы**

1. Белов С.В., Ильницкая А.В., Козьяков А.Ф. и др. Безопасность жизнедеятельности. Учебник для вузов.- М.: Высш.шк.,2005 - 448 с.
2. Безопасность жизнедеятельности / Под ред. Н.А. Белова. – М.: Знание, 2000. – 364 с.
3. Справочная книга для проектирования электрического освещения Г.Б. Кнорринга. – Л.: Энергия, 1976.
4. Robert Nystrom, Game Programming Patterns, Amazon, 2014. – 354 с.
5. Базылов К.Б., Алибаева С.А., Бабич А.А. Методические указания для студентов всех форм обучения специальности 050719 –Радиотехника электроника и телекоммуникации. – Алматы: АИЭС, - 2008. - 20 с.
6. Scott Rogers, Level Up! The Guide to Great Video Game Design, 2014. – 550 с.
7. Jason Gregory, Game Engine Architecture, 2009. – 864 с.
8. Tomas Akenine-Moller, Eric Haines, Naty Hoffman, Real-Time Rendering, 2008. – 1045 с.
9. Jesse Schell, The Art of Game Design: A book of lenses, 2008. – 520 с.
10. Raph Koster, A Theory of Fun for Game Design, 2004. - 256 с.
11. Mat Buckland, Programming Game AI By Example, 2004. – 495 с.
12. Christer Ericson, Real-Time Collision Detection, 2004. – 632 с.