

Городской научно-практический марафон «Шаги в науку»

II этап

**Секция информатики и информационных технологий**

(английский язык)

«Компьютерная игра на уроках английского языка»

*Исследовательская работа*

**Выполнено:** учеником 9 «Б» класса

МАОУ ГИМНАЗИЯ №216 «ДИДАКТ»

Куницким Дмитрием

**Руководитель:**

учитель математики

МАОУ ГИМНАЗИЯ №216 «ДИДАКТ»

Бурлакова Светлана Адамовна

---

г. Заречный, 2024

## **Содержание**

Введение.....	3
Основная часть.....	4
Этапы создания игры.....	5
Заключение.....	9
Список литературы.....	10

## ВВЕДЕНИЕ

Сегодня очень сложно представить нашу жизнь без сферы информационных технологий, они внедрились в каждый аспект нашей жизни. Они преобразуют и улучшают мир, даря нам новые возможности и перспективы развития. Как можно использовать это в учебном процессе – это и предстоит мне решить. Чаще мы увидим подростка, который с увлечением играет на компьютере или подростка, который развлекается в телефоне. Возникает вопрос, а что, если компьютерную игру использовать на уроке? Учитель иностранного языка обратилась к нам за помощью, создать игру, которая поможет в изучении лексических единиц, а конкретно в теме «Многозначные слова».

**Предмет:** приложения и продукты для создания игры.

**Объект:** программа игры «IT'SWORD»

**Целью** данной работы является создание игры для учащихся 6 класса на тему «Многозначные слова»

Для достижения поставленной цели необходимо выполнить ряд **задач:**

- 1) Изучить язык программирования C#.
- 2) Изучить Unity — среда разработки интерактивного 2D- и 3D-контента.
- 3) Понять, что такое GameDev, каково быть Back-end разработчиком.
- 4) Создать компьютерную игру, в качестве помощи в изучении многозначных слов, а в последствии освещающий и другие злободневные вопросы.

**Новизна** этой работы, на наш взгляд, заключается в том, что мы предприняли попытку создать

В работе использовались следующие **методы:**

- анализ литературы по информационным технологиям;
- практическое применение приложений и программ для создания игры.

**Практическая значимость:** при прохождении нашей игры человек получает реальную возможность улучшить внимание, концентрацию и память. Задачи, с которыми мозг еще не сталкивался, заставляют думать, рассуждать, подключать логику, визуальное восприятие и собранность, так же это ему помогает изучать английский язык, запоминать то, что он раньше не знал.

## ОСНОВНАЯ ЧАСТЬ

## **Этапы создания игры**

### **Жанр: приключения, хоррор, головоломка**

Первым делом стоит выбрать жанр — ведь от этого зависят все последующие этапы разработки. Опираясь здесь стоит на свои предпочтения и насмотренность, в нашем случае мы приняли сделать тестирование среди 6-ых классов, исходя из чего выяснили, что любимыми жанрами игр, у учеников являются- приключения и хоррор. Выбирать жанр стоит исходя из собственных возможностей. Так же, чтобы связать игру с английским языком, мы приняли решение добавить в игру жанр головоломок.

Мы выбрали приключение с элементами хоррора.

### **Платформа для разработки игры**

Любой продукт связанный с программированием невозможно создать без знания языка, в нашем случае - это C#, но и тут не обошлось без препятствий, ведь у Unity СВОЯ библиотека, которая только отдаленно связана с C#, скорее она представляет из себя почти отдельный язык, но с нотками, старого, доброго C#, под нотками я подразумеваю **некоторые**, а именно только числовые, так же малое количество методов и операторов, например: оператор if, тип переменной int и т.п.

Так же сам движок Unity- не идентична не одной из похожих программ. Это отдельная среда, с отдельной библиотекой, авторам и комьюнити.

### **Сюжет**

Главный герой мальчик Леон попадает в другую Вселенную и пытается выбраться из нее. Чтобы вернуться, ему необходимо побывать в нескольких локациях и выполнить задания. Все задания связаны с многозначностью английских слов.

Локации: Пустой город; Школа; Кукурузное поле; Деревня; Пещера; Цирк; Лес; Кладбище

### **Концепция и особенности**

Разобравшись с жанром, пора подумать и о том, чем игра будет выделяться среди конкурентов. Мы не нашли ни одной игры в жанре хоррор приключений, сделанную в 2D стиле и связанную с английским языком, в этом плане, нашей игре не с кем конкурировать.

Особенностью игры станут герои и локации, прорисованные вручную учениками 7-8 классов, сам процесс игры, то есть сценарий.

### **Перерыв**

Пройдя все эти начальные этапы разработки, стоит отложить проект на недельку и подумать о его концепции, о его логической составляющей, что бы нигде не было абсурда.

Также это время можно потратить на ознакомление с другими похожими проектами и подумать о том, что из них можно позаимствовать. Ведь внедрить какую-нибудь новую механику на этом этапе будет гораздо проще, чем во время активной фазы.

### **План разработки**

Когда кажется, что концепция и общий сценарий готовы, пора призадуматься конкретно над тем, как будет создаваться ваша собственная игра. На этом этапе уже пора составить список всех задач, которые предстоит решить и примерно прикинуть, сколько времени на это уйдет.

Составление плана включает в себя перечисление всех игровых механик, а также список ассетов, которые понадобятся для работы и другое. Ассетами в среде разработки компьютерных игр называют ресурсы, использующиеся для воспроизведения игры: текстуры, трехмерные модели персонажей и объектов и так далее.

Список может состоять из нескольких частей, каждая из которых отвечает за конкретный уровень в игре или базовые механики. Например, для одной локации- это может выглядеть так:

- Реализовать уникальную механику на этом уровне — 3 недели.
- Найти ассеты для противников (3 шт.) — 3 дня.
- Найти ассеты для снаряжения игрока на этом уровне (например, 20 мечей) — 5 дней.
- Создать качественную локацию — 2 недели.
- Расставить противников и протестировать уровень с ними — 4 дня.
- Подготовить музыку для уровня — 7 дней.
- И так далее.

Мы сделали это в общих чертах.

### **Продумывание мелочей**

На этом этапе можно задуматься и о небольших деталях при создании вашей игры. Додумать каждую мелочь в любом случае не получится, но все еще можно менять концепцию игры и добавлять к ней детали. Например, последовательность уровней, последовательность попадания снаряжения в руки игрока, особенности экономики и так далее. В нашей игре мы долго думали над последовательностью уровней, что бы можно было, их логически и по истории, связать друг с другом.

### **Черновая модель игры**

Первый прототип игры была версия 1.0. Сейчас будет выпущена под версией примерно 1.6. У нас в первой версии игры было много багов(недоработок), в дальнейшем

они были устранены, но в любом проекте есть баги, независимо от масштабов, нельзя сделать идеальный проект без багов.

### Взгляд со стороны

Чтобы создать компьютерную игру надлежащего качества, стоит еще на этапе планирования обратиться к другим компетентным людям. Разумеется, в идеальных условиях это должны быть люди из индустрии, работающие над разработкой игр уже какое-то время.

Однако если вы только начинаете делать игры, то вряд ли получится найти таких экспертов. Так что остается либо обратиться к своим друзьям и знакомым, либо пойти более экстремальным путем — начать обсуждение на форуме разработчиков игр. Форум разработчиков очень интересное место, там можно найти, как новых друзей, так и составляющие игры, например: новые механики.

### Программирование

Вот код написанный от руки в среде Unity. Он отвечает за передвижение объекта, называют такой файл с кодом скриптом, а скрипт с передвижением главного героя-PlayerMovement. Без этого скрипта не обходится почти не одна игра.

Вот сам код:

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class movePlayer : MonoBehaviour
{
    private Rigidbody2D rb;
    public float speed = 0.5f;
    private Vector2 moveVector;

    void Start()
    {
        rb = GetComponent<Rigidbody2D>();
    }

    void Update()
    {
        moveVector.x = Input.GetAxis("Horizontal");
```

```
moveVector.y = Input.GetAxis("Vertical");
rb.MovPosition (rb.position + moveVector * speed * Time.deltaTime);
}
}
```

С самого начала с припиской `using` пишутся библиотеки которые используются, обычно их не больше 10, т.к. это влияет на оптимизацию игры.

Далее идут типы переменных

Потом с припиской `void`- методы

`Rb`- это `RigidBody`- это физическое свойство объекта.

Команда `MovePosition` осуществляет перемещение игрока в пространстве.

## Движок игры

Под «движком игры» понимается технология, отвечающая за техническую сторону работы игры. То есть использование ассетов, соблюдение игровой логики, инструменты разработки и возможность запуска игры на самых разных компьютерах, телефонах и приставках — именно это задача игрового движка.

Из них наиболее известны два: `Unreal Engine` и `Unity`, причем в случае с первым нужно еще и выбрать версию движка. До сих пор сохраняет актуальность `Unreal Engine 4`, хотя индустрия уже потихоньку переходит на пятую версию, имеющую множество прорывных технологий вроде динамического реалистичного освещения. Впрочем, для работы с ним потребуется и мощный компьютер, способный обрабатывать такую графику в реальном времени.

`Unreal Engine`- очень сложный движок, новичку, да даже профессионалу без команды- не стоит туда лезть, а вот `Unity`- более подходит для команды из одного разработчика. Мы выбрали `Unity`, т.к. у нас не трипл А проект и даже не дабл А проект, у нас игра ссылается на инди, на игру малую по масштабам.

## Тестовая версия игры

После выбора движка и освоения базовых принципов работы с ним имеет смысл создать небольшой прототип игры — ту самую `Beta` версию игры. В итоге эта версия может стать как основой для будущей игры, так и тем самым первым блином, который ком. В любом случае мы получили опыт и протестировали работоспособность своей идеи, а также смогли показать игру другим. По ходу обновлений нашей игры, мы убирали баги, добавляли механики, делали оптимизацию.

## **Работаем с графикой**

Важная часть компьютерных игр — их графическая составляющая. Мы использовали векторную графику и рисовали полностью вручную.

Помимо этого, можно все-таки попытаться освоить программы для создания игровых объектов, хоть это по сути является отдельной специальностью. Весь HUD (графика, локации и герои) UI (Интерфейс) рисовали ученики 7-8 классов.

## **Записываем звук**

Большинство движков и так предлагает собственные звуки для базовых действий — ходьбы, стрельбы, ударов и так далее. Музыка также можно взять с бесплатных ресурсов, где выкладываются свободные от авторских прав треки — например, Bensound или Tunetank. Мы использовали второй.

## **Тестируем игру и дорабатываем баги**

Предположим, что вы прошли все этапы разработки игры и подготовили рабочую версию. В ней есть все заявленные механики и необходимый для их функционирования контент — в общем, тот минимум, с которым игру можно выпускать. Однако перед этим будет еще один трудоемкий этап.

Необходимо протестировать работу всех возможностей, причем во всех локациях игры и во всех возможных сценариях. Разумеется, одному человеку сделать это будет крайне затруднительно, хоть и возможно. Главное — четко фиксировать, какие вещи и на каких уровнях вы уже проверили. Например, можно вести отдельную таблицу. Недоработки- это баги, их было большое кол-во в Beta версии, по ходу обновлений большее кол-во багов было пофикшено, в альфа версии игры, их очень мало, но абсолютно любой проект не без багов.

Разумно будет раздать последнюю версию игры своим знакомым или друзьям, с которых требовать отчет. Издаем игру.

Издание игр — совершенно отдельный процесс от непосредственно разработки. В ходе этого этапа решаются вопросы размещения игры на площадках платформ, ее продвижения и рекламы.

Наша игра была загружена на: «Яндекс Диск».



Отсканируйте и скачайте нашу игру.

### **Заключение**

#### **Пошаговая инструкция создания своей компьютерной игры**

Давайте суммируем все этапы, из которых состоит разработка игры:

1. Препродакшн — определение жанра и платформ, на которых выйдет игра.
2. Создание дизайн-документа — подготовка списка механик, контента, отличительных особенностей и сюжета игры.
3. Разработка — выбор движка, реализация игровой логики, поиск и внедрение ассетов и так далее.
4. Тестирование — поиск и устранение багов.
5. Релиз — выпуск игры на обозначенных платформах и ее продвижение.
6. Поддержка — исправление найденных поклонниками недочетов и небольшие обновления, чтобы проект оставался актуальным.

#### **Подведем итоги**

В современных реалиях сделать свою игру может быть сложно, поскольку для этого сейчас не требуется даже уметь программировать, но при должных усилиях — это возможно. Настроить логику с помощью таблиц, найти бесплатные ассеты и рассказать о своем проекте как можно большему количеству людей — вот и весь рецепт успеха. Но тут есть так же есть какой-то подвох

## Рецензия на работу обучающегося 10 класса

МАОУ ГИМНАЗИЯ № 216 «ДИДАКТ»

Куницкого Дмитрия Руслановича

### «Компьютерная игра на уроках английского языка»

Рецензируемая работа «Компьютерная игра на уроках английского языка» Куницкого Дмитрия представляет собой совокупность двух проектов: написание игры на языке C++ и повышение мотивации в изучении английского языка учащихся 5-6 классов.

Поставленная в работе проблема является актуальной, так как Сегодня невозможно представить нашу жизнь без информационных технологий. Они внедрились в каждый аспект нашей повседневной жизни, они преобразуют и улучшают мир, даря нам новые возможности и перспективы развития. Как можно использовать это в учебном процессе – это и стало решением автора. Как можно компьютерную игру использовать на уроке? Этот запрос поступил от учителя иностранного языка. Методика изучения лексических единиц отработана, но вопрос повышения мотивации всегда решается каждым учителем индивидуально.

Работа соответствует требованиям, предъявляемым к проектной работе: актуальность, проблемы, предмет, объект, создание продукта, заключение. Центральным вопросом работы является повышение уровня информационной грамотности в среде детей-разработчиков, мотивации в изучении английского языка и в возможности изучения информационных технологий.

Автор проявил умения коммуникации в плане подбора команды учеников из других классов для создания графических объектов игры, информации по многозначным словам в английском языке. Процесс создания игры соответствует основным правилам: препродакшн, дизайн-документ, выбор движка, реализация игровой логики, поиск и внедрение ассетов, тестирование, релиз, поддержка — исправление найденных поклонниками недочетов и небольшие обновления, чтобы проект оставался актуальным.

В результате создания игры образовалась группа учащихся более младших классов, которые получили навыки работы в графическом редакторе, написании сценария игры и сюжета, подбора музыкальных фрагментов. Игра на уроках английского языка вызвала определенный интерес у детей с одной стороны, и

показала, что в современных условиях создать игру возможно каждому, кто готов к кропотливой работе с другой стороны.

В работе приложен QR-код, по которому можно скачать игру и запустить ее.

Практическая ценность заключается в том, что данная программа может быть использован на уроках, а также на факультативных и кружковых занятиях. Материал, представленный в этой работе, будет также полезен всем, кто интересуется вопросами написания игр.

Рецензент проекта: Бурлакова Светлана Адамовна,  
заместитель директора МАОУ ГИМНАЗИЯ № 216 «ДИДАКТ»

