

УДК 004.4
МРНТИ 14.33.09

DOI: <https://doi.org/10.37788/2023-2/149-157>

В.В. Наумов^{1*}, С.В. Балаба¹, К.С. Исканров¹, А.Б. Толегенов¹

¹Инновационный Евразийский университет, Казахстан

*(e-mail: vvn296@mail.ru)

Опыт командной разработки компьютерной игры

Аннотация

Основная проблема: в статье рассматривается опыт разработки компьютерной видеоигры командой студентов кафедры «Информационные технологии» Инновационного Евразийского университета в рамках конкурса «48h Game Making Challenge» по созданию видеоигр, проведенного в г. Павлодаре в корпусе IT-Hub Колледжа Информационных технологий. Основная информация о проведении и ссылка на форму регистрации были опубликованы на странице социальной сети Instagram @digital_drive14 и на сайте <https://digitaldrive.kz/>. В конкурсе участвовало 8 команд от разных образовательных учреждений Павлодара.

Цель: осмысление практического опыта разработки компьютерной игры.

Методы: в качестве среды и основного инструмента разработки компьютерной игры команда выбрала игровой движок Unity. Программа Unity – популярная среда разработки интерактивного 2D- и 3D-контента. Движок является бесплатным для рядового пользователя. Он был выбран во многом потому, что является многофункциональным, в нём используется компонентно-ориентированный подход, в рамках которого разработчик создает объекты и добавляет различные компоненты. Преимуществом движка является наличие огромной библиотеки ассетов и плагинов, с помощью которых можно ускорить процесс разработки игры. Графика игры была сделана в среде Photoshop. Музыка и звуки были взяты из бесплатных источников.

Результаты и их значимость: команда ИНЕУ разработала компьютерную игру «Покоритель Луны» и стала победителем конкурса «48 Hour Game Making Challenge Pavlodar 2022». В процессе разработки игры команда выбрала движок, разработала сюжет и концепцию игры, создала образ для уровня и меню. Программист разработал механику игры, написал код для управления, времени, диалогов, сообщений. Дизайнеры нарисовали окружение, задний фон, персонажи, создали все анимации для игры и нашли музыку для фона меню и самой игры.

Ключевые слова: компьютерная игра, разработка, игровой движок, конкурс.

Введение

В г. Павлодаре в рамках событийного бренда Digital Drive в корпусе IT-Hub Колледжа Информационных технологий (ул. Лермонтова, д.93) был проведен 48-часовой конкурс по созданию видеоигр. Время проведения – с 10:00 27 октября 2022г. до 10:00 29 октября 2022 г. Основная информация о проведении и ссылка на форму регистрации были опубликованы на странице социальной сети Instagram @digital_drive14 и на сайте <https://digitaldrive.kz/>.

На основании 8 поступивших заявок были сформированы 8 команд:

| № | Название команды | Состав |
|---|------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | Brainius | Татенов Азизбек Аскарлович / 20 лет / Дизайнер Гомар Мадихан Бауыржанұлы / 20 лет / Геймдизайнер Габдулкалиев Нуржан Серикович / 20 лет / Программист Нуруллаев Илья Эльдарович / 20 лет / Java-разработчик Амангельдинов Даулет Ержанович / 20 лет / Программист |
| 2 | Кванториум | Ботанов Денис Валерьевич / 21 год / Геймдизайнер-программист Гетман Алексей Сергеевич / 18 лет / Программист Ким Валентин Витальевич / 18 лет / Геймдизайнер уровней Ярмолицкая Анастасия Александровна / 22 года / Художник-дизайнер |

Продолжение

| | | |
|---|----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 3 | MALIK TEAM | БузуртановАльмалик Викторович/17 лет/программист Гоенко Иван Викторович/16 лет/геймдизайнер Панкова Людмила Олеговна/16 лет/художник Айтмухамет Анвар Аскарлович/16 лет/звукорежиссёр |
| 4 | Quantum Brake | ИксакДаянЕрболатович / 18 лет / программист Кабден Ислам Тлектесұлы / 17 лет /геймдизайнер |
| 5 | IntuCoders | Құдайберген Рамадан Жұмабекұлы/ 17 лет/ Backend-разработчик Шахан АйдынЖанжосулы/18 лет/Геймдизайнер БастемиевМураталиДастанович/18 лет/Геймдизайнер Агарков Алексей Павлович/18 лет/ Программист |
| 6 | Asdi | СарсенбаевАсылханЕрханович/16 лет/ программист АхметжанДынмухамедБолатович/16 лет/ дизайнер |
| | Improvisation | КапасоваКамилаКабыкеновна / 17 лет Григорюк Алексей Александрович / 17 лет Турсемканов Рустам Мадениетович / 16 лет ДуйсембайШыңғысЕрмеқұлы / 16 лет ЖенисАсаналиМадиярулы / 16 лет |
| | Мистическое IT | ИскаировКаиржан/18 лет/дизайнер ТолегеновАлтынбек/18 лет/саунд-дизайнер Балаба Сергей/19 лет/сценарист-программист |

26 октября 2022 г. с участниками был проведен брифинг в IT-Hub, на котором им были доведены условия проведения мероприятия:

- мероприятие проводится без заготовленного заранее арта, музыки и кода (исключение – свободно лицензируемые саунд-банки и текстуры из свободных источников; разрешены любые бесплатные игровые движки);
- во время проведения GMS48 строго запрещается мешать другим участникам, а также заимствовать идеи других команд;
- разрешается продолжать работу во внеурочное время, однако при прибытии на место проведения необходимо продемонстрировать прогресс в разработке;
- запрещается оскорблять реальных лиц, разжигать межнациональную и межрелигиозную вражду;
- возрастной рейтинг разрабатываемой игры не должен превышать рейтинг PG;
- игровой мир должен содержать ассеты с логотипами «Ditigal Drive» и «48h Game Making Challenge» (логотипы можно перекрашивать под основную палитру игры);
- основная палитра игры выбирается случайным образом из заранее заготовленных наборов цветов.

Призовой фонд 48h GMS составил 1 000 000 тенге и распределен следующим образом:

- 1 место – 500 000 тенге;
- 2 место – 300 000 тенге;
- 3 место – 200 000 тенге.

«Приз зрительских симпатий» - 100 000 тенге.

27 октября 2022г. в 10:00 по основному месту проведения было проведено открытие 48h GMS, а также проведена жеребьевка (вытягивание случайных листов) среди команд на цветовую палитру, обязательную локацию и обязательного персонажа из заготовленных вариантов.

По результатам жеребьевки были определены следующие критерии:

- Brainius – лес, ковбой;
- Кванториум – горы, приведение;
- Malik Team – степь, мастер боевых искусств;
- Quantum Brake – средневековый замок, волшебник;
- Intu Coders – пустыня, сыщик;
- Asdi – средневековый замок, мастер боевых искусств;
- Improvisation – космос, робот;
- Мистическое IT – луна, волшебник.

29 октября 2022г. в 10:00 команды сдали свои работы, после чего было проведено закрытие мероприятия, на котором участникам было доведено, что им необходимо прибыть на место проведения Pavlodar Digital Day 2022 (ГДК им. Естая), где в 12:00 будет проведено голосование среди посетителей мероприятия на номинацию «Приз зрительских симпатий» и в 13:00 будут оглашены победители.

Собранные работы были переданы жюри в составе:

- Амангельды Галымжан, бизнес-аналитик CSI;
- Нурканов Канат, арт-директор GlitchAnimationLab;
- Анаев Айдос, художник по текстурам SaberInteractive;
- Кусаинов Аяз, старший программист TKR.

29.10.2022г. в 13:00 жюри были объявлены победители:

1 место – «Мистическое IT»;

2 место – «Brainius»;

3 место – «Кванториум»;

Приз зрительских симпатий – «Brainius».

Таким образом, победителем конкурса «48 Hour Game Making Challenge Pavlodar 2022» стала команда «Мистическое IT», состоящая из студентов группы АиР21-202 кафедры «Информационные технологии» ИнЕУ: Балаба Сергей, Искаиров Каиржан, Толегенов Алтынбек.

Материалы и методы

Перед разработкой игры были созданы необходимые условия: локация, персонаж и цветовая гамма. Команде для разработки игры была поставлена задача – сделать локацию луны и добавить туда персонаж волшебника. Из всех вариантов игровых движков команда выбрала движок Unity, так как он наиболее подходил для поставленной задачи и, кроме того, команда была знакома с языком программирования C# [1]. Графика игры была сделана в среде Photoshop, был использован 8-битный стиль, как в старых играх [2]. Музыка и звуки были взяты из бесплатных источников.

Результаты

Команда ИнЕУ разработала игру «Покоритель Луны» и стала победителем конкурса «48 Hour Game Making Challenge Pavlodar 2022». В этом конкурсе участвовало 8 команд от разных образовательных учреждений.

Обсуждение

Сначала команда выбрала движок, разработала сюжет и концепцию игры, создала образ для уровня и меню. Далее программист разработал механику игры, написал код для управления, времени, диалогов, сообщений. Дизайнеры нарисовали окружение, задний фон, персонажи, доделали все анимации для игры и нашли музыку для фона меню и самой игры.

Движок Unity. Программа Unity – популярная среда разработки интерактивного 2D- и 3D-контента. Такие программы, как Unity, называют движками. Разработала её компания Unity Technologies в 2005 году. Движок бесплатный для рядового пользователя. Он является многофункциональным, в нем используется компонентно-ориентированный подход, в рамках которого разработчик создает объекты (например, главного героя), к ним добавляет различные компоненты, например, визуальное отображение персонажа и способы управления им. Преимуществом движка являлось наличие огромной библиотеки ассетов и плагинов, с помощью которых можно значительно ускорить процесс разработки игры.

Сюжет игры. Сюжет игры рассказывает про смелого космонавта, который решил покорить Луну. Но его мечте было не суждено сбыться, так как он встретил волшебника, который уже покорил Луну. Волшебник решает наказать космонавта, сослав его на Луну. Теперь герою нужно будет найти путь к своей ракете и вернуться домой. Так же было решено добавить альтернативную концовку, чтобы игрок мог сам определить, какая для него лучшая. Меню сделано так, чтобы оно подсказывало одну из возможных концовок игры.

Геймплей. Управление состоит из клавиш «Стрелочки» или «"W", "A", "S", "D"» и пробел. Одна из механик игры это «Zoom» или «Отдаление» работает как отдаление камеры, чтобы увеличить обзор видимости (рисунок 1).



Рисунок 1 – Пример работы «Zoom»

Также в игре присутствует таймер в верхнем левом углу. Этот таймер связан с вариантом концовки игры. Если игрок не успеет за определенное время дойти до финиша, то обрушится метеоритный дождь.

По всей локации расставлены ловушки, которые будут мешать дойти до цели (рисунок 2).

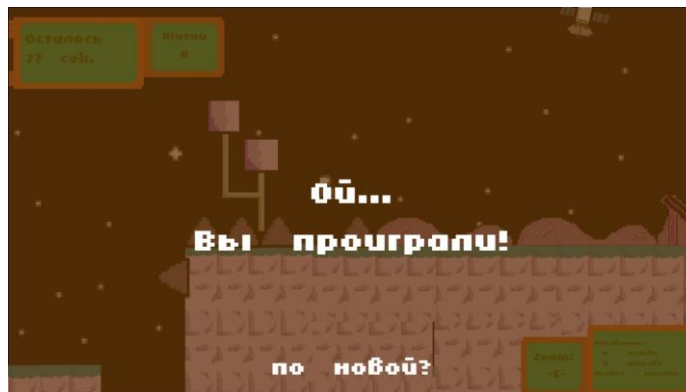


Рисунок 2 – Экран проигрыша

Для большего представления об игре на рисунке 3 представлены фотографии некоторых моментов игры.

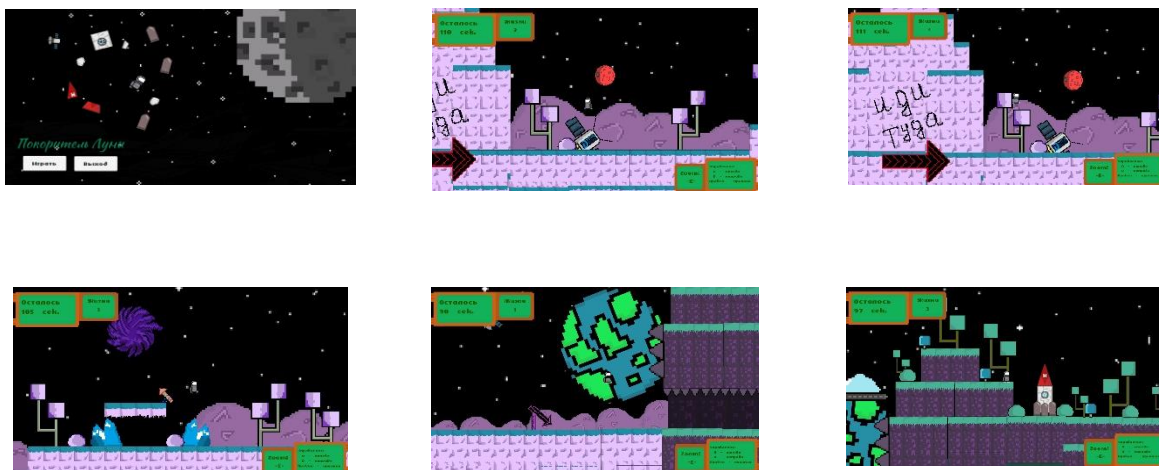


Рисунок 3 – Моменты игры

Некоторые фрагменты кода программы. Рассмотрим фрагменты кода, служащие для перемещения игрока в плоскости и слежения камеры за персонажем.

Код 1. Движение персонажа:

```
public class Player : MonoBehaviour
{
```

```
public Rigidbody2D rb;
public Animator anim;
public SpriteRenderer sp;

public Vector2 moveV;
public float speed;
public float jump;
public int health;
public float numHealth;
public Text text;

public GameObject Plays;

public AudioSource audi;
public AudioSource audi2;
public AudioSource dead;

public GameObject PanelDead;

public float upDubbleJump;

public bool onGround;
public Transform groundCheck;
public float checkRadius = 0.5f;
public LayerMask Ground;

public bool faceRight = true;
private int jumps = 0;
public int maxjump = 2;

void Start()
{
    rb = GetComponent<Rigidbody2D>();
    anim = GetComponent<Animator>();
    p = GetComponent<SpriteRenderer>();
}

void Update()
{
    moveV.x = Input.GetAxis(«Horizontal»)*Time.deltaTime;
    anim.SetFloat(«moveX», Mathf.Abs(moveV.x));
    rb.velocity = new Vector2(moveV.x * speed, rb.velocity.y);

    if ((moveV.x > 0 && !faceRight) || (moveV.x < 0 && faceRight))
    {
        transform.localScale *= new Vector2(-1, 1);
        faceRight = !faceRight;
    }

    onGround = Physics2D.OverlapCircle(groundCheck.position, checkRadius, Ground);
    anim.SetBool(«onGround», onGround);

    if (Input.GetKeyDown(KeyCode.Space))
    {
        if (onGround)
        {
            audi.Play();
        }
    }
}
```

```

rb.AddForce(Vector2.up * jump);
}
else if (++jumps < maxjump)
{
audi2.Play();
rb.velocity = new Vector2(0, upDubbleJump);
}

}
if (onGround)
{
jumps = 0;
}
text.text = health.ToString();
if (health <= 0)
{
PanelDead.SetActive(true);
Plays.GetComponent<Player>().enabled = false;
dead.Play();

}

if (Input.GetKeyDown(KeyCode.Escape))
{
Debug.Log(«Меню»);

SceneManager.LoadScene(«MainMenu»);
}
}

```

Эта часть обеспечивает логику управления: когда нажата левая или правая кнопка на клавиатуре, персонаж начинает движение:

```

moveV.x = Input.GetAxis(«Horizontal»)*Time.deltaTime;
anim.SetFloat(«moveX», Mathf.Abs(moveV.x));
rb.velocity = new Vector2(moveV.x * speed, rb.velocity.y);

```

Здесь же есть метод, который проверяет, в какую сторону смотрит персонаж. Он нужен для облегчения анимации, чтобы не вводить игрока в заблуждение о том, почему его персонаж исполняет лунную походку:

```

if ((moveV.x > 0 && !faceRight) || (moveV.x < 0 && faceRight))
{
transform.localScale *= new Vector2(-1, 1);
faceRight = !faceRight;
}

```

Далее идет проверка земли. Когда персонаж касается земли, срабатывает проверка. Это нужно для того, чтобы правильно работала анимация движения:

```

onGround = Physics2D.OverlapCircle(groundCheck.position, checkRadius, Ground);
anim.SetBool(«onGround», onGround);

```

Данная проверка служит также и для логики прыжка, чтобы игрок не прыгал только по одной плоскости. В коде есть проверка на количество прыжков и высоту:

```

if (Input.GetKeyDown(KeyCode.Space))
{
if (onGround)
{
audi.Play();
rb.AddForce(Vector2.up * jump);
}
else if (++jumps < maxjump)
{
audi2.Play();
rb.velocity = new Vector2(0, upDubbleJump);
}
}
if (onGround)
{
jumps = 0;
}
}

```

В этой части рассмотрим метод жизни, которые за счет игровых ситуаций будут отниматься, а по окончании будет проигрываться окно поражения:

```

text.text = health.ToString();

if (health <= 0)
{
PanelDead.SetActive(true);
Plays.GetComponent<Player>().enabled = false;
dead.Play();
}

```

Код 2. Слежение камеры за игроком:

```

public class FollowCam : MonoBehaviour {
public Transform target;
public float smoothTime = 0.2f;

private Vector3 _velocity = Vector3.zero;

void Start() {

}

void FixedUpdate() {

Vector3 targetPosition = new Vector3(target.position.x, target.position.y,
transform.position.z);
transform.position = Vector3.SmoothDamp(transform.position, targetPosition, ref _velocity,
smoothTime);
}
}

```

В коде используется метод Vector3, который служит для правильной навигации за персонажем, то есть система координат X, Y и Z:

```
Vector3 targetPosition = new Vector3(target.position.x, target.position.y, transform.position.z);
```

Следующая строчка служит для обеспечения плавности передвижения камеры:

```
transform.position = Vector3.SmoothDamp(transform.position, targetPosition, ref _velocity, smoothTime).
```

Заключение

Участие студентов в данном конкурсе показал эффективность командной разработки компьютерной игры. Преимущество такой работы заключается в многозадачности, которая достигается в процессе спланированной координации действий и творческого поиска решений.

Такая работа позволяет создать объективные условия, создаваемые для разработчиков игр в реальном производстве. Это способствует формированию у студентов базовых компетенций в области моделирования игры, её оформления и апробации.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Хокинг Джозеф. Unity в действии. Мультиплатформенная разработка на C#. – СПб.: Изд-во Питер, 2022. – 448 с.
- 2 Дронов В.А. Графика и анимация. – СПб.: БХВ-Петербург, 2006. – 656 с.

REFERENCES

- 1 Khoking, D. (2022). Unity v deystvii. Multiplatformennaya razrabotka na C# [Unity in action. Multiplatform development in C#]. Sankt-Peterburg: Izd. Piter [in Russian].
- 2 Dronov, V.A. (2006). Grafika i animaciya [Graphics and Animation]. Sankt-Peterburg: BHV-Peterburg [in Russian].

В.В. Наумов^{1*}, С.В. Балаба¹, К.С. Искаиров¹, А.Б. Төлегенов¹
¹Инновациялық Еуразия университеті, Қазақстан

Компьютерлік ойындарды командамен әзірлеудің тәжірибесі

Мақалада 2022 жылғы 27-29 қазанда Павлодарда Ақпараттық технологиялар колледжінің IT-Hub корпусында (Лермонтов көшесі, 93-үй) өткізілген «48h game Making Challenge» 48 сағаттық байқауы шеңберінде Инновациялық Еуразия университетінің «Ақпараттық технологиялар» кафедрасы студенттерінің командасының бейне ойындар жасау тәжірибесі қарастырылады. Өткізу туралы негізгі Ақпарат және тіркеу формасына сілтеме Instagram әлеуметтік желісінің @digital_drive14 бетінде және сайтта жарияланды <https://digitaldrive.kz/>. Байқауға Павлодардың түрлі білім беру мекемелерінен 8 команда қатысты.

Мақаланың мақсаты – компьютерлік ойындарды әзірлеудің практикалық тәжірибесін пайымдау. Компьютерлік ойынды дамытудың негізгі құралы және құралы ретінде команда Unity ойын қозғалтқышын таңдады. Unity бағдарламасы интерактивті 2D және 3D мазмұнын дамытудың танымал ортасы болып табылады. Қозғалтқыш қарапайым пайдаланушы үшін мүлдем тегін. Unity қозғалтқышы да таңдалды, өйткені ол көп функциялы және компоненттерге бағытталған тәсілді қолданады, оның аясында әзірлеуші нысандарды жасайды және оларға әртүрлі компоненттерді қосады. Қозғалтқыштың артықшылығы, сонымен қатар ойынның даму үрдісін едәуір жылдамдатуға болатын ассеттер мен плагиндердің үлкен кітапханасының болуы. Ойынның графикасы Photoshop ортасында жасалды. Музыка мен дыбыстар тегін көздерден алынды.

ИнЕУ командасы «Айды бағындырушы» компьютерлік ойынын әзірлеп, «48 сағаттық ойын Making Challenge Pavlodar 2022» байқауының жеңімпазы атанды. Ойынды дамыту барысында команда алдымен қозғалтқышты таңдап, ойынның сюжеті мен тұжырымдамасын жасады, деңгей мен мәзір үшін кескін жасады. Әрі қарай, бағдарламашы ойын механикасын әзірледі, басқару, уақыт, диалог, хабарламалар үшін код жазды. Дизайнерлер қоршаған ортаны, фонды, кейіпкерлерді бояды, ойнауға арналған барлық анимацияларды жасады және мәзірдің фонына және ойынның өзіне музыка тапты.

Түйінді сөздер: компьютерлік ойын, даму, ойын қозғалтқышы, байқау.

V.V. Naumov^{1*}, S.V. Balaba¹, K.S. Iskairov¹, A.B. Tolegenov¹
¹Innovative University of Eurasia, Kazakhstan

Experience in team computer game development

The article discusses the experience of developing a computer video game by a team of students of the Department of Information Technology of the Innovative University of Eurasia within the 48-hour competition «48h Game Making Challenge» for creating video games, held in Pavlodar on

October 27-29, 2022 in the IT-Hub building of the College of Information Technologies (93 Lermontov St.). Basic information about the event and a link to the registration form were published on the Instagram @digital_drive14 social network page and on the website <https://digitaldrive.kz/>. 8 teams from different educational institutions of Pavlodar participated in the competition.

The purpose of the article is to comprehend the practical experience of developing a computer game. The team chose the Unity game engine as the environment and the main tool for developing a computer game. The Unity program is a popular development environment for interactive 2D and 3D content. The engine is completely free for the average user. The Unity engine was also chosen because it is multifunctional and uses a component-oriented approach, in which the developer creates objects and adds various components to them. The advantage of the engine was also the presence of a huge library of assets and plugins, with which you can significantly speed up the game development process. The graphics of the game were made in Photoshop. The music and sounds were taken from free sources.

InEU team developed a computer game «Conqueror of the Moon» and became the winner of the contest «48 Hour Game Making Challenge Pavlodar 2022». During the development of the game, the team first chose the engine, developed the plot and concept of the game, created an image for the level and menu. Then the programmer developed the mechanics of the game, wrote code for control, time, dialogues, messages. The designers drew the environment, background, characters, created all the animations for the game and found music for the background of the menu and the game itself.

Keywords: computer game, development, game engine, competition.

Дата поступления рукописи в редакцию: 24.04.2023 г.