

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ  
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»  
(БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и  
автоматизированных систем

## **КУРСОВОЙ ПРОЕКТ**

по дисциплине: **Основы программирования**  
по теме: “Разработка видеоигры в жанре Экономический симулятор”

Автор работы \_\_\_\_\_ Игнатъев Артур Олегович ПВ-223  
(подпись)

Руководитель проекта \_\_\_\_\_ Черников Сергей Викторович  
(подпись)

Оценка \_\_\_\_\_

Белгород 2023

## Содержание

Введение .....	3
1. Про выбранный жанр и игровой движок .....	4
1.1. Жанр “Экономический симулятор” .....	4
1.2. Игровой движок Ren’Py. ....	4
2. Суть игры .....	6
3. Этапы создания проекта.....	7
4. Описание основных алгоритмов ивентов .....	8
5. Информационный экран и базовые события. ....	10
6. Кнопки управления и экраны королевством .....	14
7. Заключение .....	19
Список источников и литературы .....	20

## Введение

Разработка видеоигр представляет собой значительную сферу экономики, в которой участвует огромное количество людей, используются сложные технологии. Кроме того, видеоигры стали важной частью жизни подростков и детей. Эта курсовая работа была создана для более глубокого понимания процесса разработки видеоигр.

Объектом исследования является процесс разработки видеоигры.

Предметом исследования является игровой движок Ren'Py.

Цель данной работы – разработка видеоигры в жанре “Экономический симулятор”.

Для достижения данной цели были поставлены следующие задачи:

1. Изучить игровой движок Ren'Py.
2. Разработать видеоигру в жанре “Экономический симулятор”.

## 1. Про выбранный жанр и игровой движок

### 1.1. Жанр Экономический симулятор.

Экономический симулятор - это жанр видеоигр, целью которых является моделирование и управление экономическими процессами. В этих играх игроки принимают роль предпринимателя, менеджера или владельца бизнеса, стремясь развивать свои экономические империи.

Основными элементами экономических симуляторов являются создание и управление финансами, развитие инфраструктуры, управление ресурсами и принятие стратегических решений для достижения финансового успеха. Игры этого жанра могут охватывать различные отрасли, такие как сельское хозяйство, производство, финансы, торговля и даже управление городом.

Игроки сталкиваются с вызовами, такими как управление бюджетом, анализ рынка, улучшение производственных процессов и конкуренция с виртуальными конкурентами. Успех зависит от умения игрока эффективно распределять ресурсы, предвидеть изменения в рыночных условиях и принимать обдуманные стратегические решения.

Экономические симуляторы пользуются популярностью среди игроков, которые наслаждаются аспектами стратегического планирования и управления, а также теми, кто интересуется миром бизнеса и экономики. Эти игры могут предоставить не только развлечение, но и обучение финансовым и стратегическим навыкам.

### 1.2. Игровой движок Ren'Py.

Ren'Py — это бесплатный кроссплатформенный движок для цифрового повествования. Он позволяет легко комбинировать слова, изображения и звуки для создания визуальных новелл и игр-симуляторов жизни.

Язык сценариев Ren'Py позволяет легко писать визуальные романы и другие игры, требующие большого количества написания. Его легко освоить, и он хорошо масштабируется для самых крупных проектов. Даже без настройки Ren'Py предоставляет те функции, которые игроки привыкли ожидать от своих визуальных новелл.

Однако, Ren'Py использует язык программирования Python, поэтому его можно использовать не только для визуальных романов и симуляторов жизни, а и для других игр различных жанров.

#### Достоинства и недостатки

- + Кроссплатформенность, поддерживаемые платформы: Windows, macOS, Android, iOS;
- + Простой язык сценариев;
- + Легко настроить и локализовать;
- + Ren'Py может использовать преимущества аппаратного ускорения на поддерживаемых компьютерах, а в случае отсутствия поддержки использовать программное обеспечение;
- + Ren'Py совместим с ресурсами различных популярных форматов;
- + Простота освоения.
- + Бесплатный и открытый исходный код
- + Хорошая поддержка
- Ограниченные возможности для 3D-графики;
- Не подходит для сложных геймплейных механик.
- Хотя Ren'Py поддерживает экспорт на мобильные устройства, иногда возникают трудности с оптимизацией и адаптацией для различных устройств.
- Требуется навыков программирования языка Python

Ren'Py был выбран в силу простоты его освоения и показать, что его можно использовать, не только для создания визуальных новелл.

## 2. Суть игры

Суть игры состоит в следующем: игроку предстоит управлять экономикой королевства, судьбами граждан и добываемыми ресурсами. Во время прохождения игроку могут встречаться случайные события, а именно: голодомор, чума, нападение разбойников, плодородный сезон, повышенная рождаемость. Конечная цель игры набрать 1000 золотых монет.

### 3. Этапы создания проекта

Выделяют разное количество этапов для создания как сайтов, так и игр, поэтому оптимальным решением будет выбор 4 необходимых этапов.

#### 1. Определение целей разработки

На данном шаге была принята идея создания экономической игры с элементами торговли

#### 2. Разработка технического задания (ТЗ)

Составление правильного ТЗ просто необходимый шаг в этапах создания любого проекта.

Требованиями к игре являются:

1. использование Ren'Py;
2. создание визуально понятного интерфейса;
3. не сложная проходимость.

#### 3. Создание и программирование основы проекта

На данном шаге был создан проект на основе Ren'Py.

Помимо самого использования самого Ren'Py был необходим язык программирования Python.

Были продуманы основная логика и механики игры.

#### 4. Тестирование

В данном этапе проводилось тестирование всевозможно-доступных игровых механик и геймплейных особенностей.

#### 4. Описание основных алгоритмов ивентов

Ивент “Голодомор” – представляет собой большее потребление еды населением.

```
# Ивент "Голодомор" (плохое событие)
if (random_events == 1):
    food -= random.randint(2, 5) * (peasants + farmers + artisans + warriors)
```

Листинг 1. Ивент “Голодомор”

Ивент “Чума” – случайные смерти групп населения в зависимости от населённости.

```
# Ивент "Чума" (плохое событие)
if (random_events == 2):
    for y in range(10, 100, 10):
        if(y <= peasants <= y + 9 or y <= farmers <= y + 9 or y <= artisans <= y + 9 or y <= warriors <= y + 9):
            for n in range(10, 100, 10):
                if(n <= peasants <= n + 9):
                    x = n // 10 + 1
                    random_plague = random.randint(x, x * 5)
                    peasants -= random_plague
                    died += random_plague
                if(n <= farmers <= n + 9):
                    x = n // 10 + 1
                    random_plague = random.randint(x, x * 5)
                    farmers -= random_plague
                    died += random_plague
                if(n <= artisans <= n + 9):
                    x = n // 10 + 1
                    random_plague = random.randint(x, x * 5)
                    artisans -= random_plague
                    died += random_plague
                if(n <= warriors <= n + 9):
                    x = n // 10 + 1
                    random_plague = random.randint(x, x * 5)
                    warriors -= random_plague
                    died += random_plague
```

Листинг 2. Ивент “Чума”

Ивент “Нападение бандитов” – представляет собой нападение на население королевства.

Сначала на себя удар принимают воины (1 воин = 1 бандиту), но, если их не хватает, после их смерти принимают удар ремесленники (2 ремесленника = 1 бандиту). После смерти ремесленников под удар попадают фермеры и крестьяне (3 фермера или крестьянина = 1 бандиту)



```
# Ивент "Нападение бандитов" (плохое событие)
if (random_events == 3):
    bandits = random.randint((turns // 10) + 1, (turns // 5) + 1)
    if (bandits >= warriors):
        if (warriors > 0):
            bandits -= warriors
            warriors = 0
        if (artisans > 0 and bandits > 0):
            bandits_temp = bandits
            bandits = ((2 * bandits) - artisans) // 2
            artisans -= 2 * bandits_temp
            if (artisans < 0):
                artisans = 0
        if (farmers > 0 and bandits > 0):
            bandits_temp = bandits
            bandits = ((3 * bandits) - farmers) // 3
            farmers -= 3 * bandits_temp
            if (farmers < 0):
                farmers = 0
        if (peasants > 0 and bandits > 0):
            bandits_temp = bandits
            bandits = ((3 * bandits) - peasants) // 3
            peasants -= 3 * bandits_temp
            if (peasants < 0):
                peasants = 0
    elif (bandits == warriors):
        warriors -= bandits
    elif (bandits <= warriors):
        warriors -= bandits
```

Листинг 3. Ивент “Нападение бандитов”

Ивент “Плодородный сезон” – представляет собой увеличенный прирост еды.

```
# Ивент "Плодородный сезон" (хорошее событие)
if (random_events == 4):
    food += farmers * random.randint(10, 15)
```

Листинг 4. “Плодородный сезон”

Ивент “Повышенная рождаемость” – представляет собой увеличенный прирост рождённых.

```
# Ивент "Повышенная рождаемость" (хорошее событие)
if (random_events == 5):
    born = 0
    random_born = random.randint(3, 10)
    peasants += random_born
    born += random_born
```

Листинг 5. “Повышенная рождаемость”

## 5. Информационный экран и базовые события

screen open\_info\_panel – кнопка открытия панели ресурсов.

screen info\_panel – панель ресурсов. В ней можно увидеть:

- 1 монеты
- 2 дерево
- 3 камень
- 4 железо
- 5 еда
- 6 крестьяне
- 7 фермеры
- 8 ремесленники
- 9 воины
- 10 рождённых за ход
- 11 умерших за ход
- 12 количество сделанных ходов

```
# Панель ресурсов
screen info_panel:
    modal True
    frame:
        xalign 0.01
        yalign 0.05
        hbox:
            vbox:
                add "info_panel/coin_05d.png"
                add "info_panel/wood_01a.png"
                add "info_panel/stoneblock_01a.png"
                add "info_panel/ingot_01c.png"
                add "info_panel/food.png"
                add "info_panel/boots_01a.png"
                add "info_panel/gloves_01a.png"
                add "info_panel/helmet_01a.png"
                add "info_panel/helmet_02b.png"
                add "info_panel/heart.png"
                add "info_panel/skull_01a.png"
            vbox:
                text "[coins]" size 40
                text "[wood]" size 40
                text "[stone]" size 40
                text "[iron]" size 40
                text "[food]" size 40
```

```

text "[peasants]" size 40
text "[farmers]" size 40
text "[artisans]" size 40
text "[warriors]" size 40
text "[born]" size 40
text "[died]" size 40
text "Ход: [turns]" size 40
textbutton "Закреть":
    action [
        Hide("info_panel"), # Скрыть open_info_panel
        Show("open_info_panel"), # Показать панель info_panel
        Show("open_kingdom_management"), # Показать панель kingdom_management
        Show("open_trade_panel"),
        Show("complete_the_move")
    ]

screen open_info_panel:
    frame:
        xalign 0.01
        yalign 0.05
        textbutton "Ресурсы королевства":
            action [
                Hide("open_info_panel"), # Скрыть open_info_panel
                Hide("open_kingdom_management"), # Скрыть open_kingdom_management
                Hide("complete_the_move"),
                Hide("open_trade_panel"),
                Show("info_panel") # Показать панель info_panel
            ]

```

Листинг 6. Кнопка открытия панели ресурсов и панель ресурсов

screen complete\_the\_move – кнопка пропуска хода.

label move – выполнений пропуска хода в котором происходят ивенты и обычные действия такие как:

- 1 случайное выпадение ивента
- 2 люди едят за ход
- 3 смерть
- 4 прирост еды
- 5 рождение
- 6 добыча ресурсов

```

screen complete_the_move:
    frame:
        xalign 0.99
        yalign 0.05
        textbutton "Завершить ход" action Call("move")

label move:

```

```

python:
    turns += 1
    import random
    random_events = random.randint(1, 25)

    # Люди едят за ход
    if (random_events != 1):
        food -= peasants + farmers + 2 * artisans + 3 * warriors

    # Ивент "Голодомор" (плохое событие)
    if (random_events == 1):
        food -= random.randint(2, 5) * (peasants + farmers + artisans + warriors)

    # Смерть
    died = 0
    if (random_events != 2):
        for y in range(10, 100, 10):
            if(y <= peasants <= y + 9 or y <= farmers <= y + 9 or y <= artisans <= y + 9 or y <= warriors <= y + 9):
                for n in range(10, 100, 10):
                    if(n <= peasants <= n + 9):
                        x = 0
                        random_plague = random.randint(x, x + 1)
                        peasants -= random_plague
                        died += random_plague
                    if(n <= farmers <= n + 9):
                        x = 0
                        random_plague = random.randint(x, x + 1)
                        farmers -= random_plague
                        died += random_plague
                    if(n <= artisans <= n + 9):
                        x = 0
                        random_plague = random.randint(x, x + 1)
                        artisans -= random_plague
                        died += random_plague
                    if(n <= warriors <= n + 9):
                        x = 0
                        random_plague = random.randint(x, x + 0)
                        warriors -= random_plague
                        died += random_plague

    # Ивент "Чума" (плохое событие)
    if (random_events == 2):
        for y in range(10, 100, 10):
            if(y <= peasants <= y + 9 or y <= farmers <= y + 9 or y <= artisans <= y + 9 or y <= warriors <= y + 9):
                for n in range(10, 100, 10):
                    if(n <= peasants <= n + 9):
                        x = n // 10 + 1
                        random_plague = random.randint(x, x * 5)
                        peasants -= random_plague
                        died += random_plague
                    if(n <= farmers <= n + 9):
                        x = n // 10 + 1
                        random_plague = random.randint(x, x * 5)
                        farmers -= random_plague
                        died += random_plague
                    if(n <= artisans <= n + 9):
                        x = n // 10 + 1
                        random_plague = random.randint(x, x * 5)
                        artisans -= random_plague
                        died += random_plague
                    if(n <= warriors <= n + 9):
                        x = n // 10 + 1

```

```

        random_plague = random.randint(x, x * 5)
        warriors -= random_plague
        died += random_plague

# Ивент "Нападение бандитов" (плохое событие)
if (random_events == 3):
    bandits = random.randint((turns // 10) + 1, (turns // 5) + 1)
    if (bandits >= warriors):
        if (warriors > 0):
            bandits -= warriors
            warriors = 0
        if (artisans > 0 and bandits > 0):
            bandits_temp = bandits
            bandits = ((2 * bandits) - artisans) // 2
            artisans -= 2 * bandits_temp
            if (artisans < 0):
                artisans = 0
        if (farmers > 0 and bandits > 0):
            bandits_temp = bandits
            bandits = ((3 * bandits) - farmers) // 3
            farmers -= 3 * bandits_temp
            if (farmers < 0):
                farmers = 0
        if (peasants > 0 and bandits > 0):
            bandits_temp = bandits
            bandits = ((3 * bandits) - peasants) // 3
            peasants -= 3 * bandits_temp
            if (peasants < 0):
                peasants = 0
    elif (bandits == warriors):
        warriors -= bandits
    elif (bandits <= warriors):
        warriors -= bandits

# Прирост еды
if (random_events != 4):
    food += farmers * random.randint(3, 7)

# Ивент "Плодородный сезон" (хорошее событие)
if (random_events == 4):
    food += farmers * random.randint(10, 15)

# Рождение
if (random_events != 5):
    born = 0
    random_born = random.randint(0, 1)
    peasants += random_born
    born += random_born

# Ивент "Повышенная рождаемость" (хорошее событие)
if (random_events == 5):
    born = 0
    random_born = random.randint(3, 10)
    peasants += random_born
    born += random_born

# Добыча ресурсов
if (artisans > 0):
    wood += random.randint(1, 5) * artisans
    stone += random.randint(1, 3) * artisans
    iron += 1 * artisans

```

Листинг 7. кнопка пропуска хода, выполнений пропуска хода.

## 6. Кнопки управления и экраны королевством

`screen open_kingdom_management` – кнопка открытия панели управления королевством.

`screen kingdom_management` – панель управления королевством. В ней есть кнопки:

- 1 отправить на ферму
- 2 вернуть из фермы
- 3 отправить в мастерскую
- 4 вернуть из мастерской
- 5 отправить в казарму
- 6 вернуть из казармы
- 7 закрыть

```
# Панель управления королевством
screen kingdom_management:
    modal True
    frame:
        xalign 0.4
        yalign 0.05
        hbox:
            vbox:
                textbutton "Отправить на ферму" action Call("farm")
                textbutton "Вернуть из фермы" action Call("farm_back")
                textbutton "Отправить в мастерскую" action Call("workshop")
                textbutton "Вернуть из мастерской" action Call("workshop_back")
                textbutton "Отправить в казарму" action Call("barracks")
                textbutton "Вернуть из казармы" action Call("barracks_back")
                null height 15
                textbutton "Закрыть":
                    action [
                        Hide("kingdom_management"), # Скрыть kingdom_management
                        Show("open_info_panel"), # Показать панель info_panel
                        Show("open_kingdom_management"), # Показать панель open_kingdom_management
                        Show("complete_the_move"),
                        Show("open_trade_panel")
                    ]

# Кнопка открытия панель управления королевством
screen open_kingdom_management:
    frame:
        xalign 0.4
        yalign 0.05
        textbutton "Управление королевством":
            action [
```

```

        Hide("open_info_panel"), # Скрыть open_info_panel
        Hide("open_kingdom_management"), # Скрыть open_kingdom_management
        Hide("complete_the_move"),
        Hide("open_trade_panel"),
        Show("kingdom_management") # Показать панель kingdom_management
    ]

# Отправить на ферму
label farm:
    python:
        if (coins >= 2 and peasants >= 1):
            coins -= 2
            peasants -= 1
            farmers += 1
        else:
            "У вас недостаточно монет или крестьян."
        return

# Вернуть из фермы
label farm_back:
    python:
        if (farmers >= 1):
            farmers -= 1
            peasants += 1
        else:
            "У вас недостаточно фермеров."
        return

# Отправить в мастерскую
label workshop:
    python:
        if (coins >= 5 and peasants >= 1):
            coins -= 5
            peasants -= 1
            artisans += 1
        else:
            "У вас недостаточно монет или крестьян."
        return

# Вернуть из мастерской
label workshop_back:
    python:
        if (artisans >= 1):
            artisans -= 1
            peasants += 1
        else:
            "У вас недостаточно ремесленников."
        return

# Отправить в казарму
label barracks:
    python:
        if (coins >= 10 and peasants >= 1):
            coins -= 10
            peasants -= 1
            warriors += 1
        else:
            "У вас недостаточно монет или крестьян."
        return

# Вернуть из казармы
label barracks_back:

```

```
python:
    if (warriors >= 1):
        warriors -= 1
        peasants += 1
    else:
        "У вас недостаточно воинов."
    return
```

Листинг 8. Кнопка открытия панели управления королевством и панель управления королевством.

screen open\_trade\_panel – кнопка открытия панели торговли.

screen trade\_panel – панель торговли. Она содержит кнопки:

- 1 продать дерево
- 2 купить дерево
- 3 продать камень
- 4 купить камень
- 5 продать железо
- 6 купить железо
- 7 закрыть

```
screen trade_panel:
    modal True
    frame:
        xalign 0.7
        yalign 0.05
        hbox:
            vbox:
                textbutton "Продать дерево" action Call("sell_wood")
                textbutton "Купить дерево" action Call("buy_wood")
                textbutton "Продать камень" action Call("sell_stone")
                textbutton "Купить камень" action Call("buy_stone")
                textbutton "Продать железо" action Call("sell_iron")
                textbutton "Купить железо" action Call("buy_iron")
                null height 15
                textbutton "Закрыть":
                    action [
                        Hide("trade_panel"), # Скрыть kingdom_management
                        Show("open_kingdom_management"), # Показать open_kingdom_management
                        Show("open_info_panel"), # Показать панель info_panel
                        Show("open_kingdom_management"), # Показать панель open_kingdom_management
                        Show("complete_the_move")
                    ]

# Кнопка открытия панель управления торговлей
screen open_trade_panel:
    frame:
```



```

xalign 0.7
yalign 0.05
textbutton "Торговля":
    action [
        Hide("open_info_panel"), # Скрыть open_info_panel
        Hide("open_kingdom_management"), # Скрыть open_kingdom_management
        Hide("kingdom_management"), # Скрыть панель open_kingdom_management
        Hide("complete_the_move"),
        Show("trade_panel")
    ]

# Продать дерево
label sell_wood:
    python:
        if (wood >= 15):
            coins += 1
            wood -= 15
        else:
            "У вас недостаточно дерева."
        return

# Купить дерево
label buy_wood:
    python:
        if (coins >= 1):
            coins -= 1
            wood += 10
        else:
            "У вас недостаточно монет."
        return

# Продать камень
label sell_stone:
    python:
        if (stone >= 10):
            coins += 1
            stone -= 10
        else:
            "У вас недостаточно камня."
        return

# Купить камень
label buy_stone:
    python:
        if (coins >= 1):
            coins -= 1
            stone += 5
        else:
            "У вас недостаточно монет."
        return

# Продать железо
label sell_iron:
    python:
        if (iron >= 5):
            coins += 1
            iron -= 5
        else:
            "У вас недостаточно железа."
        return

# Купить железо

```

```
label buy_iron:
    python:
        if (coins >= 1):
            coins -= 1
            iron += 1
        else:
            "У вас недостаточно монет."
    return
```

## 7. Заключение

В ходе курсовой работы были изучены основы игрового движка Ren'Py.

Была реализована разработка игры в жанре “Экономическая стратегия” при помощи игрового движка Ren'Py.

Таким образом, все поставленные задачи были выполнены, цель работы достигнута.

## Список источников и литературы

1. Ren'Py Manual [электронный ресурс] Режим доступа:  
<https://www.renpy.org/doc/html/>
2. Python 3 Manual [электронный ресурс] Режим доступа:  
<https://docs.python.org/3/>
3. Itch.io Assets [электронный ресурс] Режим доступа:  
<https://itch.io/game-assets>