Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

Специальность 09.02.07 Информационные системы и программирование

**ОТЧЕТ**

**ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ**

Профессиональный модульПМ.02 Осуществление интеграции

программных модулей

*(наименование профессионального модуля)*

Выполнил:

обучающийся учебной группы № ИСИП-519

Пастушенко Р. Ю

*(И.О. Фамилия)*

Проверил:

руководитель практики от колледжа:

Аксёнова Т. Г

*(И.О. Фамилия)*

**Москва**

**20­­­­23**

**Содержание**

[**Введение** 3](#_Toc132131576)

[**Глава 1. Разработка ПО** 4](#_Toc132131577)

[1.1 Скрипт для телефона 4](#_Toc132131578)

[1.2 Скрипт для говорящих персонажей 9](#_Toc132131579)

[1.3 Небольшие скрипты 11](#_Toc132131580)

[**ГЛАВА 2. ТЕСТИРОВАНИЕ ПО** 15](#_Toc132131581)

[Тестовый пример #1: 15](#_Toc132131582)

[Тестовый пример #2: 16](#_Toc132131583)

[Тестовый пример #3: 16](#_Toc132131584)

[**ЗАКЛЮЧЕНИЕ** 17](#_Toc132131585)

[**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ** 18](#_Toc132131586)

# **Введение**

В последнее время дети-взрослые стали намного меньше читать книг.

Раньше дети намного больше времени проводили время за чтением какой-либо книги, мы считаем, что это огромная проблема в наше время.

Регулярное чтение позволяет лучше формулировать собственные мысли. Благодаря книгам появляется возможность расширить круг общения, стать интересным собеседником и начать получать удовольствие от взаимодействия с миром. Не говоря уже о том, что прочитанное положительно влияет на наше сознание, шлифует речь и формирует собственный взгляд на вещи. Обеспечить полноценную работу мозга на долгие годы.

По сей день, дети или взрослые зачастую проводят больше времени за какой-либо компьютерной игрой, и тем самым тратят большее количество времени на что-то бесполезное.

Актуальность разрабатываемой игры связанна, с соединением игры и чтением книг. Читатель нашей игры, получит тот же спектр возможностей как у книги, одновременно расслабляясь от прохождения игры.

# **Глава 1. Разработка ПО**

## 1.1 Скрипт для телефона

Ниже представлен код, благодаря которому происходит симуляции сообщений в телефоне. В данном коде задаем размеры, координаты разных объектов, так же прописываем анимацию и звуки сообщений.

define nvl\_mode = "phone"

define MC\_Name = "Гоша"

init -1 python:

phone\_position\_x = 0.3

phone\_position\_y = 0.5

def Phone\_ReceiveSound(event, interact=True, \*\*kwargs):

if event == "show\_done":

renpy.sound.play("audio/ReceiveText.ogg")

def Phone\_SendSound(event, interact=True, \*\*kwargs):

if event == "show\_done":

renpy.sound.play("audio/SendText.ogg")

def print\_bonjour():

print("bonjour")

transform phone\_transform(pXalign=0.5, pYalign=0.5):

xcenter pXalign

yalign pYalign

transform phone\_appear(pXalign=0.5, pYalign=0.5):

xcenter pXalign

yalign pYalign

on show:

yoffset 1080

easein\_back 1.0 yoffset 0

transform message\_appear(pDirection):

alpha 0.0

xoffset 50 \* pDirection

parallel:

ease 0.5 alpha 1.0

parallel:

easein\_back 0.5 xoffset 0

transform message\_appear\_icon():

zoom 0.0

easein\_back 0.5 zoom 1.0

transform message\_narrator:

alpha 0.0

yoffset -50

parallel:

ease 0.5 alpha 1.0

parallel:

easein\_back 0.5 yoffset 0

screen PhoneDialogue(dialogue, items=None):

style\_prefix "phoneFrame"

frame at phone\_transform(phone\_position\_x, phone\_position\_y):

if len(dialogue) == 1:

at phone\_appear(phone\_position\_x, phone\_position\_y)

viewport:

draggable True

mousewheel True

yinitial 1.0

vbox:

null height 20

use nvl\_phonetext(dialogue)

null height 100

screen nvl\_phonetext(dialogue):

style\_prefix None

$ previous\_d\_who = None

for id\_d, d in enumerate(dialogue):

if d.who == None:

text d.what:

xpos -335

ypos 0.0

xsize 350

text\_align 0.5

italic True

size 28

slow\_cps False

id d.what\_id

if d.current:

at message\_narrator

else:

if d.who == MC\_Name:

$ message\_frame = "phone\_send\_frame.png"

else:

$ message\_frame = "phone\_received\_frame.png"

hbox:

spacing 10

if d.who == MC\_Name:

box\_reverse True

if previous\_d\_who != d.who:

if d.who == MC\_Name:

$ message\_icon = "phone\_send\_icon.png"

else:

$ message\_icon = "phone\_received\_icon.png"

add message\_icon:

if d.current:

at message\_appear\_icon()

else:

null width 107

vbox:

yalign 1.0

if d.who != MC\_Name and previous\_d\_who != d.who:

text d.who

frame:

padding (20,20)

background Frame(message\_frame, 23,23,23,23)

xsize 350

if d.current:

if d.who == MC\_Name:

at message\_appear(1)

else:

at message\_appear(-1)

text d.what:

pos (0,0)

xsize 350

slow\_cps False

if d.who == MC\_Name :

color "#FFF"

text\_align 1.0

xpos -580

else:

color "#000"

id d.what\_id

$ previous\_d\_who = d.who

style phoneFrame is default

style phoneFrame\_frame:

background Transform("phone\_background.png", xcenter=0.5,yalign=0.5)

foreground Transform("phone\_foreground.png", xcenter=0.5,yalign=0.5)

ysize 815

xsize 495

style phoneFrame\_viewport:

yfill True

xfill True

yoffset -20

style phoneFrame\_vbox:

spacing 10

xfill True

Скрипт для работы телефона

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1. Инициализация персонажей в телефоне

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 2. Часть с диалогом

Изображение выглядит как текст, электроника, снимок экрана

Автоматически созданное описание

Рисунок 3. Работа телефона

1.2 Скрипт для говорящих персонажей

Данный скрипт создает визуальную часть того, что персонаж говорит, когда код видит, что сейчас будет речь заданного нами персонажа он подсвечивается, а противоположный человек затухает, прописываем анимацию, цвет, на сколько персонаж будет подсвечиваться, на сколько затемнятся, прописываем зум персонажа, и задаем условия для выполнения скрипта.

define sprite\_focus = {}

default speaking\_char = None

transform sprite\_highlight(sprite\_name):

    function SpriteFocus(sprite\_name)

init -10 python:

    import math

    def name\_callback(event, interact=True, name=None, \*\*kwargs):

        global speaking\_char

        if event == "begin":

            speaking\_char = name

    class SpriteFocus(object):

        def \_\_init\_\_(self, char\_name):

            self.char\_name = char\_name

        def \_\_call\_\_(self, trans, start\_time, anim\_time):

            def get\_ease(t):

                return .5 - math.cos(math.pi \* t) / 2.0

            global sprite\_focus, speaking\_char

            char\_name = self.char\_name

            if char\_name not in sprite\_focus:

                sprite\_focus[char\_name] = False

            anim\_length = 0.2

            bright\_change = 0.08

            sat\_change = 0.2

            zoom\_change = 0.0025

            y\_change = 1

            is\_talking = speaking\_char == char\_name

            if isinstance(sprite\_focus[char\_name], (int, float)) and anim\_time < sprite\_focus[char\_name]:

                sprite\_focus[char\_name] = is\_talking

            if sprite\_focus[char\_name] != is\_talking and isinstance(sprite\_focus[char\_name], bool):

                sprite\_focus[char\_name] = anim\_time

                if renpy.is\_skipping() or renpy.in\_rollback():

                    sprite\_focus[char\_name] = is\_talking

            curr\_time = max(anim\_time - sprite\_focus[char\_name],0)

            curr\_ease = 1.0

            if curr\_time < anim\_length and not isinstance(sprite\_focus[char\_name], bool):

                curr\_ease = get\_ease(curr\_time/anim\_length)

            else:

                sprite\_focus[char\_name] = is\_talking

            if is\_talking:

                trans.matrixcolor = SaturationMatrix((1.0-sat\_change) + curr\_ease \* sat\_change) \* BrightnessMatrix(-bright\_change + curr\_ease \* bright\_change)

                trans.zoom = min(curr\_ease \* zoom\_change + (1.0-zoom\_change), 1.0)

                trans.yoffset = y\_change - curr\_ease \* y\_change

            else:

                trans.matrixcolor = SaturationMatrix(1.0 - curr\_ease \* sat\_change) \* BrightnessMatrix(curr\_ease \* -bright\_change)

                trans.zoom = max(1.0 - curr\_ease \* zoom\_change, (1.0-zoom\_change))

                trans.yoffset = y\_change \* curr\_ease

            return 0

Скрипт позволяющий выделять говорящих персонажей

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 4. Создание копий героев игры

## 1.3 Небольшие скрипты



Рисунок 5. Вывод героев

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 6. Анимация глаз

Изображение выглядит как экран, изображение, темный, облака

Автоматически созданное описание

Рисунок 7. Закрытие глаз

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 7. Выбор действий, и переход на выбранную часть

Изображение выглядит как текст, Веб-сайт

Автоматически созданное описание

Рисунок 8. Выбор действий

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 9. Прописываем курсор и музыку в игре

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 10. Код для интерактивной карты

Задаем координаты кнопок, прописываем размер, прописываем нажатие, прописываем переход на другой лейбл при нажатии на кнопку. Так же при наведении на кнопку, задаем подсказку

Изображение выглядит как текст, в помещении, шаг, окно

Автоматически созданное описание

Рисунок 11. Подсветка кнопки, и подсказка

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 11. Инициализация персонажей

# **ГЛАВА 2. ТЕСТИРОВАНИЕ ПО**

**Аннотация теста**

|  |  |
| --- | --- |
| **Название проекта** |  |
| **Рабочая версия** | 1.0 |
| **Имя тестирующего** | Пастушенко Роман Юрьевич |
| **Дата(ы) теста** | 01.04.2023 |

## Тестовый пример #1:

|  |  |
| --- | --- |
| **Тестовый пример #** | 01 |
| **Приоритет тестирования** | Высокий |
| **Заголовок/название теста** | Телефон |
| **Краткое изложение теста** | Вывод текста в окно телефона |
| **Этапы теста** | Дойти до сюжетного этапа |
| **Тестовые данные** | - |
| **Ожидаемый результат** | Вывод сообщения в телефоне со звуковым сигналом |
| **Фактический результат** | Вывод сообщения в телефоне со звуковым сигналом |
| **Статус** | Зачет |
| **Предварительное условие** | Запустить игру |
| **Постусловие** | Нет |
| **Примечания/комментарии** | Нет |

## Тестовый пример #2:

|  |  |
| --- | --- |
| **Тестовый пример #** | 02 |
| **Приоритет тестирования** | Высокий |
| **Заголовок/название теста** | Выбор действия |
| **Краткое изложение теста** | Предоставить возможность выбора действий игроку |
| **Этапы теста** | Запустить игру |
| **Тестовые данные** | Выбрать предлагаемые сюжетные линии |
| **Ожидаемый результат** | Вывод действий |
| **Фактический результат** | Вывод действий |
| **Статус** | Зачет |
| **Предварительное условие** | Запустить игру |
| **Постусловие** | Нет |
| **Примечания/комментарии** | Нет |

## Тестовый пример #3:

|  |  |
| --- | --- |
| Тестовый пример **#** | 03 |
| **Приоритет тестирования** | Высокий |
| **Заголовок/название теста** | Интерактивная карта |
| **Краткое изложение теста** | Выбор передвижение на картинке |
| **Этапы теста** | Запустить игру |
| **Тестовые данные** | Выбор стрелочки ↑ ↓ |
| **Ожидаемый результат** | После нажатия на кнопку, переход на другой лейбл |
| **Фактический результат** | После нажатия на кнопку, переход на другой лейбл |
| **Статус** | Зачет |
| **Предварительное условие** | Запустить игру |
| **Постусловие** | Нет |
| **Примечания/комментарии** | Нет |

# **ЗАКЛЮЧЕНИЕ**

Выполняя задание от практики, нам удалось реализовать огромную часть нашего диплома. Было создано большое количеств скриптов, оживляющие игру, позволяя пользователю почувствовать разные мелочи игры. Так же доработали меню игры, улучили звуковую сопровождающую игры. Дописали, и проработали сюжетные линии, сделали расхождения в сюжете. Выполнении огромную работу с фоном игры, сделали его более качественным и подходящем к вселенной игры. Данная игра позволит пройти мини путешествие от лица главного героя, и изменять сюжетные линии благодаря выбору действий.

# **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. <https://www.renpy.org/wiki/renpy/rus/doc/tutorials/Руководство_для_начинающих>. – Руководство по ren.py
2. [https://docs.huihoo.com/renpy/6.12.0/quickstart.html - Руководство по ren.py](https://docs.huihoo.com/renpy/6.12.0/quickstart.html%20-%20Руководство%20по%20ren.py)
3. <https://metanit.com/python/tutorial/> - Руководсво по python
4. <https://gimp.su/kak-pol-zovat-sya-gimp/> Руководсво gimp
5. Altlinux: Discord – [Электронный ресурс]. – URL: <https://www.altlinux.org/Discord>
6. Википедия. Microsoft Word – [Электронный ресурс]. – URL: https://ru.wikipedia.org/wiki/Microsoft\_Word