Министерство образования Московской области

Государственное образовательное учреждение высшего образования Московской области

«Государственный гуманитарно-технологический университет»

**Ликино-Дулевский политехнический колледж – филиал ГГТУ**

**КУРСОВОЙ ПРОЕКТ**

Тема: Разработка программного модуля «Визуальная новела»

МДК 01.01 «Разработка программных модулей»

**Выполнила:**

Подставкина Виолетта

студентка группы ИСП.20А

09.02.07 Информационные системы и программирование

очной формы обучения

**Руководитель:**

Кузьмина Елена Евгеньевна

Оценка

\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

Подпись руководителя

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ликино-Дулево

2023 год

Оглавление

[1. Введение 3](#_Toc133912854)

[2. Назначение разработки 3](#_Toc133912855)

[2.1. Требования к программному модулю 3](#_Toc133912856)

[Требования к функциональным характеристикам: 3](#_Toc133912857)

[2.1. Требования к составу и параметрам технических средств: 4](#_Toc133912858)

[2.2. Требования к информационной и программной совместимости: 4](#_Toc133912859)

[3. Руководство программиста 4](#_Toc133912860)

[3.2. Разработка программного модуля: 6](#_Toc133912861)

[4. Руководство пользователя 14](#_Toc133912862)

[4.1. Выполнение программного модуля: 14](#_Toc133912863)

[Для пользователя: 14](#_Toc133912864)

[5. Тестирование программного модуля 17](#_Toc133912865)

[6. Заключение 18](#_Toc133912866)

[7. Список литературы: 19](#_Toc133912867)

[8. Приложение 20](#_Toc133912868)

1. Введение

Визуальная новела - это жанр компьютерных игр, который сочетает в себе элементы интерактивной книги и анимационного фильма. Основной акцент в этом жанре делается на сюжет и персонажей, а не на игровой механике и геймплее. Визуальные новеллы обычно содержат большое количество текста, иллюстраций, анимации и звуковых эффектов, которые помогают создать атмосферу и погрузить игрока в историю. Игрок может взаимодействовать с персонажами, делать выборы, которые влияют на развитие сюжета и

концовку игры.

2. Назначение разработки

Разработка проводится на основании приказа о курсовом проектировании.

Приложение должно выполнять следующие функции:

* Сохранение игрового процесса
* Загрузка существующего сохранения
* Работающий функционал
* Параметры изменения звука
* Помощь игроку в игровом интерфейсе

2.1. Требования к программному модулю

## Требования к функциональным характеристикам:

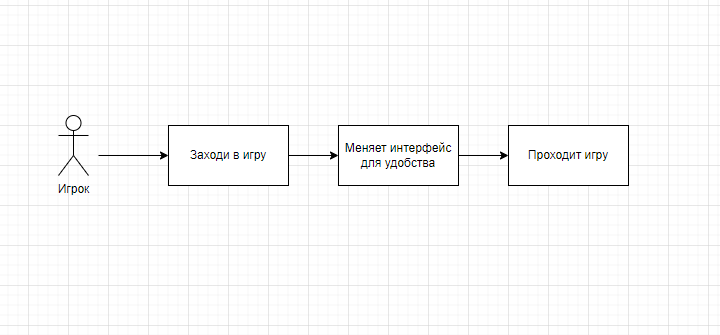


Рис.1 «ER-Диаграмма»

Ключевые функциональные требования к системе управления заданиями:

1. Изменение интерфейса и параметров игры

2. Прохождение игры используя весь ее функционал

## 2.1. Требования к составу и параметрам технических средств:

Минимальные характеристики компьютера

|  |  |
| --- | --- |
| Процессор | Ryzen 5 2600 |
| Память | 16 ГБ |
| Разрешение экрана | От 1920x1080 |
| Устройства ввода | Клавиатура, мышь |

Оптимальный характеристики компьютера

|  |  |
| --- | --- |
| Процессор | Ryzen 5 2600 |
| Память | От 16 ГБ |
| Разрешение экрана | От 1920x1080 |
| Устройства ввода | Клавиатура, мышь |

2.2. Требования к информационной и программной совместимости:

Для корректной работы программы необходимо:

Операционная система: Windows, Linux, MacOs.

3. Руководство программиста

Ren'Py - это свободная и бесплатная среда разработки визуальных новелл, созданная на языке программирования Python. Она предоставляет разработчикам множество инструментов для создания интерактивных книг и игр, включая возможность добавления анимации, звуков и музыки. Ren'Py также позволяет легко создавать персонажей и задавать им диалоги, а также создавать ветвящиеся сюжеты с различными концовками в зависимости от выборов игрока. Благодаря своей простоте и гибкости, Ren'Py стал одним из самых популярных инструментов для создания визуальных новелл.

3.1. Дерево проекта:

Папка gui - хранит в себе элементы главного меню игры

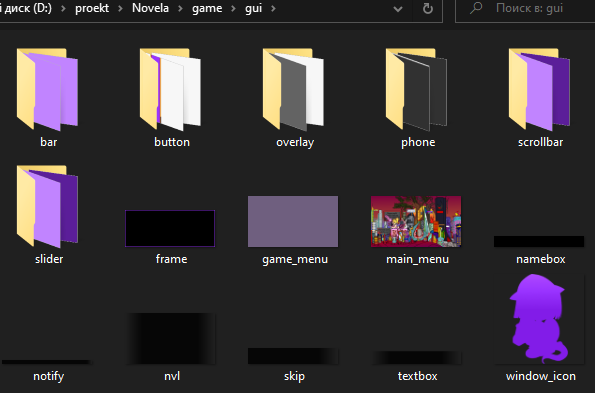


Рис. 2 «Содержимое папки gui»

Папка images – хранит в себе элементы используемые в смой игре. Персонажи и фоны

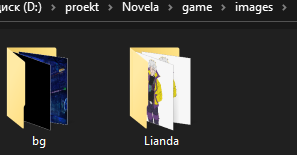


Рис. 3 «Содержимое папки images»

Папка music – хранит в себе фоновую музыку которая используется в игре

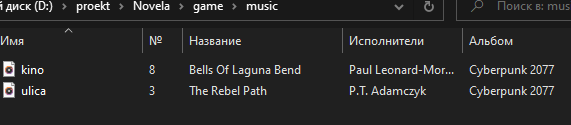


Рис. 4 «Содержимое папки music»

Папка saves – хранит в себе внутренние сохранения игры

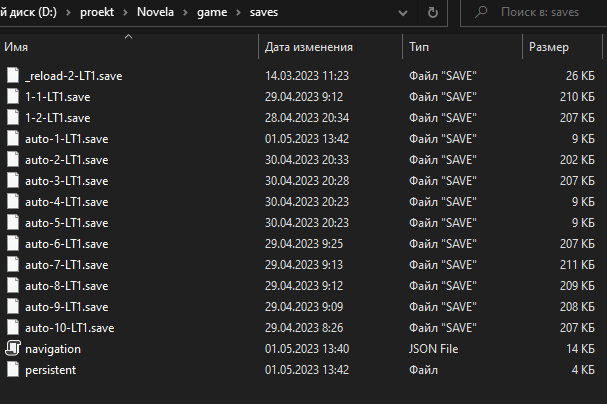


Рис. 5 «Содержимое папки saves»

Папка sounds – хранит в себе звуки, которые используются в игре

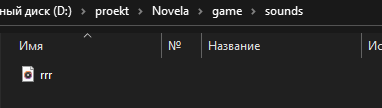


Рис. 6 «Содержимое папки sounds»

## 3.2. Разработка программного модуля:

Создание проекта:

После того как мы скачали renpy, открываем саму программу



Рис. 7 «Окно главного экрана программы»

В папках слева мы можем попасть:

* Папка в которой находится игра – game
* Папка с названием игры и всем ее содержимым – base
* Папка с элементами для игры, фоном и персонажами – image
* Папка со звук, который используется в игре – звук
* Папка с элементами интерфейса – gui

В папках справа мы можем попасть:

* Папка со скриптом игры – script.rpy
* Папка с опциями игры – options.rpy
* Папка с интерфейсом игры где можно изменять цвет текста и т.п. – gui.rpy
* Папка с созданием внутриигрового интерфейса – screens.rpy

Когда игрок запустит игру он попадет на главное меню, где он может начать игру, загрузить сохранение, настроить игру, узнать о ее создании, посмотреть игровые кнопки, и выйди из игры



Рис. 8 «Главное меню игры»

## Экран игрового меню #########################################################  
##  
## Всё это показывает основную, обобщённую структуру экрана игрового меню. Он  
## вызывается с экраном заголовка и показывает фон, заголовок и навигацию.  
##  
## Параметр scroll может быть None, или "viewport", или "vpgrid", когда этот  
## экран предназначается для использования с более чем одним дочерним экраном,  
## включённым в него.  
  
screen game\_menu(title, scroll=None, yinitial=0.0):  
  
style\_prefix "game\_menu"  
  
if main\_menu:  
add gui.main\_menu\_background  
else:  
add gui.game\_menu\_background  
  
frame:  
style "game\_menu\_outer\_frame"  
  
hbox:  
  
## Резервирует пространство для навигации.  
frame:  
style "game\_menu\_navigation\_frame"  
  
frame:  
style "game\_menu\_content\_frame"  
  
if scroll == "viewport":  
  
viewport:  
yinitial yinitial  
scrollbars "vertical"  
mousewheel True  
draggable True  
pagekeys True  
  
side\_yfill True  
  
vbox:  
transclude  
  
elif scroll == "vpgrid":  
  
vpgrid:  
cols 1  
yinitial yinitial  
  
scrollbars "vertical"  
mousewheel True  
draggable True  
pagekeys True  
  
side\_yfill True  
  
transclude  
  
else:  
  
transclude  
  
use navigation  
  
textbutton \_("Вернуться"):  
style "return\_button"  
  
action Return()  
  
label title  
  
if main\_menu:  
key "game\_menu" action ShowMenu("main\_menu")  
  
  
style game\_menu\_outer\_frame is empty  
style game\_menu\_navigation\_frame is empty  
style game\_menu\_content\_frame is empty  
style game\_menu\_viewport is gui\_viewport  
style game\_menu\_side is gui\_side  
style game\_menu\_scrollbar is gui\_vscrollbar  
  
style game\_menu\_label is gui\_label  
style game\_menu\_label\_text is gui\_label\_text  
  
style return\_button is navigation\_button  
style return\_button\_text is navigation\_button\_text  
  
style game\_menu\_outer\_frame:  
bottom\_padding 45  
top\_padding 180  
  
background "gui/overlay/game\_menu.png"  
  
style game\_menu\_navigation\_frame:  
xsize 420  
yfill True  
  
style game\_menu\_content\_frame:  
left\_margin 60  
right\_margin 30  
top\_margin 15  
  
style game\_menu\_viewport:  
xsize 1380  
  
style game\_menu\_vscrollbar:  
unscrollable gui.unscrollable  
  
style game\_menu\_side:  
spacing 15  
  
style game\_menu\_label:  
xpos 75  
ysize 180  
  
style game\_menu\_label\_text:  
size gui.title\_text\_size  
color gui.accent\_color  
yalign 0.5  
  
style return\_button:  
xpos gui.navigation\_xpos  
yalign 1.0  
yoffset -45

Загрузки:

В окне загрузки находятся слоты для сохранения игры на любом ее этапе, так же тут можно загрузить сохранение

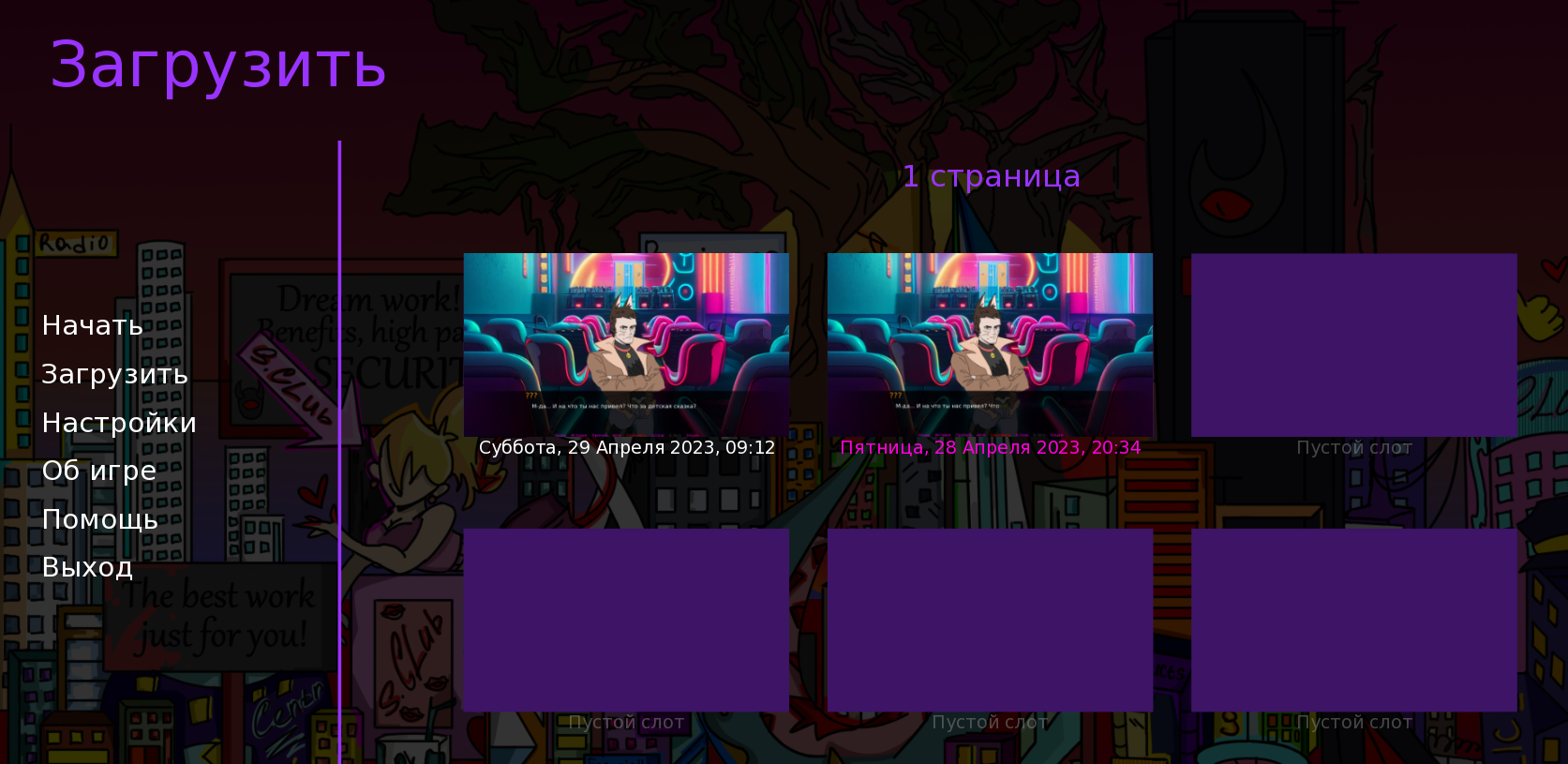


Рис. 9 «Окно загрузки»

define config.save\_directory = "Novela-1674496280"

define config.after\_load\_transition = None

 ## Кнопка слота сохранения.

define gui.slot\_button\_width = 414

define gui.slot\_button\_height = 309

define gui.slot\_button\_borders = Borders(15, 15, 15, 15)

define gui.slot\_button\_text\_size = 21

define gui.slot\_button\_text\_xalign = 0.5

define gui.slot\_button\_text\_idle\_color = gui.idle\_small\_color

define gui.slot\_button\_text\_selected\_idle\_color = gui.selected\_color

define gui.slot\_button\_text\_selected\_hover\_color = gui.hover\_color

screen save():

tag menu

use file\_slots(\_("Сохранить"))

screen load():

tag menu

use file\_slots(\_("Загрузить"))

screen file\_slots(title):

default page\_name\_value = FilePageNameInputValue(pattern=\_("{} страница"), auto=\_("Автосохранения"), quick=\_("Быстрые сохранения"))

use game\_menu(title):

Настройки:

В окне настроек находятся элементы для изменения текста, музыки, экрана

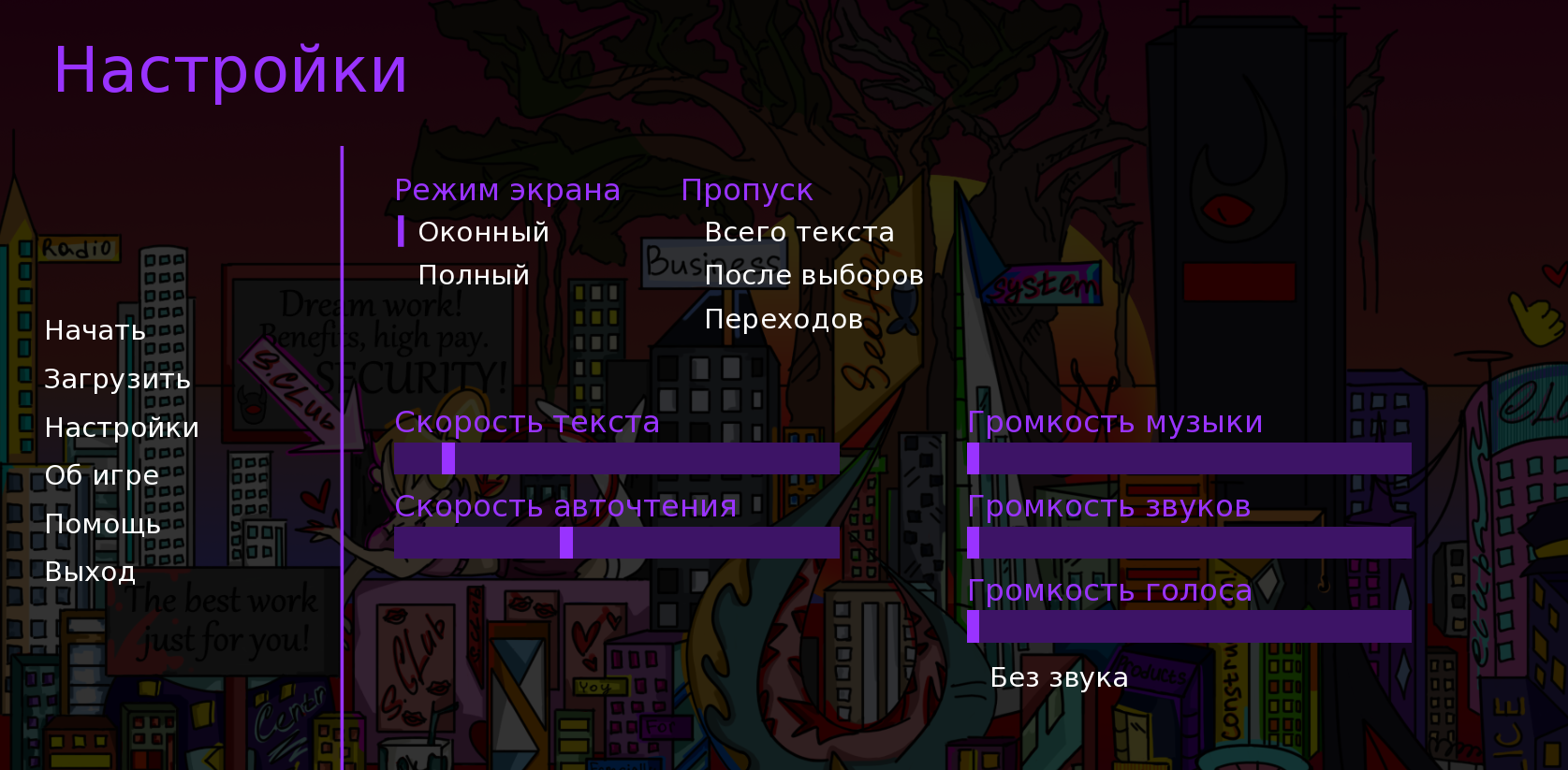


Рис.10 «Окно настройки»

screen preferences():

tag menu

use game\_menu(\_("Настройки"), scroll="viewport"):

vbox:

hbox:

box\_wrap True

if renpy.variant("pc") or renpy.variant("web"):

vbox:

style\_prefix "radio"

label \_("Режим экрана")

textbutton \_("Оконный") action Preference("display", "window")

textbutton \_("Полный") action Preference("display", "fullscreen")

vbox:

style\_prefix "check"

label \_("Пропуск")

textbutton \_("Всего текста") action Preference("skip", "toggle")

textbutton \_("После выборов") action Preference("after choices", "toggle")

textbutton \_("Переходов") action InvertSelected(Preference("transitions", "toggle"))

Помощь:

В окне помощи можно узнать играбельные кнопки

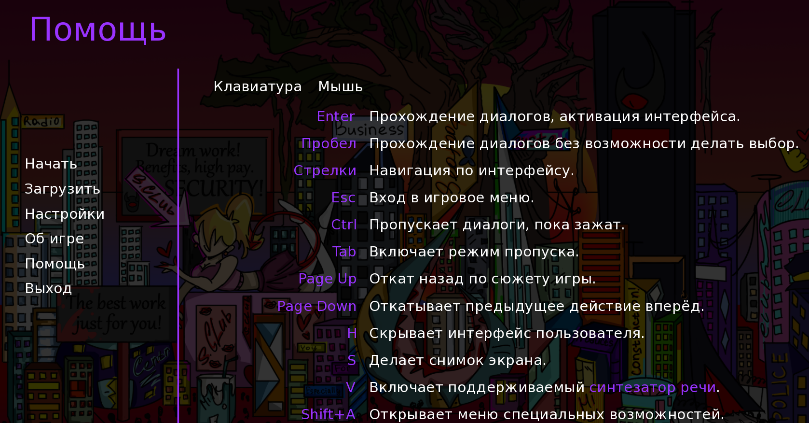


Рис. 11 «Окно помощи»

screen help():

tag menu

default device = "keyboard"

use game\_menu(\_("Помощь"), scroll="viewport"):

style\_prefix "help"

vbox:

spacing 23

hbox:

textbutton \_("Клавиатура") action SetScreenVariable("device", "keyboard")

textbutton \_("Мышь") action SetScreenVariable("device", "mouse")

if GamepadExists():

textbutton \_("Геймпад") action SetScreenVariable("device", "gamepad")

if device == "keyboard":

use keyboard\_help

elif device == "mouse":

use mouse\_help

elif device == "gamepad":

use gamepad\_help

Внутриигровой экран:

Нажав на кнопку «Начать» начинается игра



Рис. 12 «Внутриигровое окно, момент с выбором ответа»

menu:

"Тогда уж давай на каких-нибудь покемонов лучше!"

"Ну пойдем":

jump pokemon

return

label pokemon:

show der z

play sound zlo

d "НЕ НУ ТЫ СОВСЕМ ЧТО ЛИ?"

d "Это был сарказм! Я пошутил!"

show der ned

d "Совсем уже... Не надо мне еще такого счастья"

d "Так то на дворе 2123 год!"

d "А ты Хэк, привел меня на это старье..."

4. Руководство пользователя

4.1. Выполнение программного модуля:

Для пользователя:

Для того чтобы воспользоваться представленной программой, необходимо произвести следующие действия: зайти в программу renpy 🡪 запустить проект

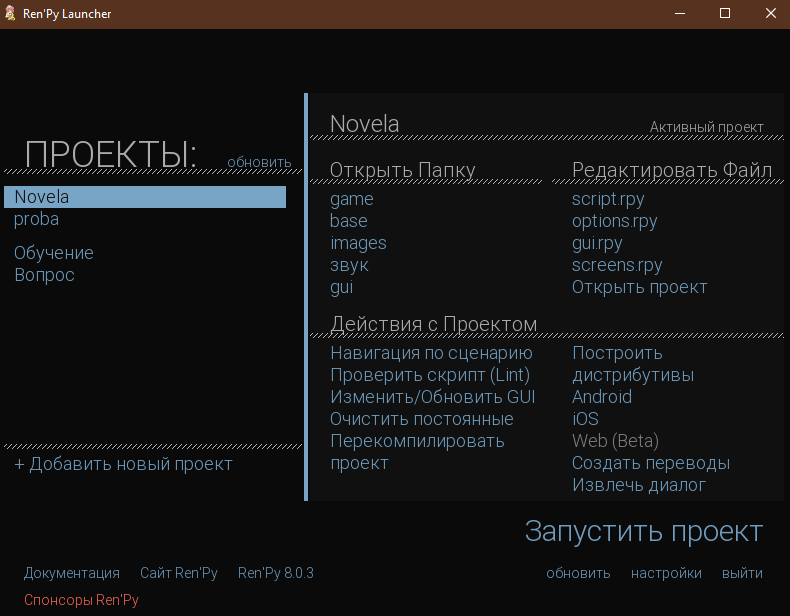


Рис. 13 «Окно запуска проекта»

После запуска проекта, перед нами появляется главное меню игры



Рис. 14 «Главное меню игры»

Так как для каждого пользователя, будут характерны свои настройки игры, для начала зайдем в настройки. В настройках меняем те или иные параметры для игры, которые нам нужны



Рис.15 «Настройки игры»

Чтобы ознакомиться с управлением игры, нужно зайти в помощь. Тут мы можем посмотреть за что отвечает та или иная кнопка на клавиатуре или мыши

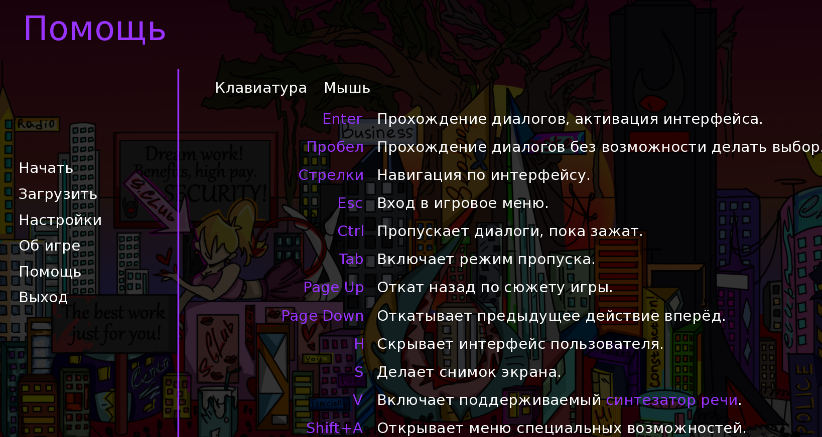


Рис. 16 «Окно помощи»

Для того чтобы начать игру, нажимаем начать. После этого нас перенесет в саму игру, где после того как мы завершим игру, нас выкинет в главное меню, где мы можем начать игру заново или загрузить нужное нам сохранение.

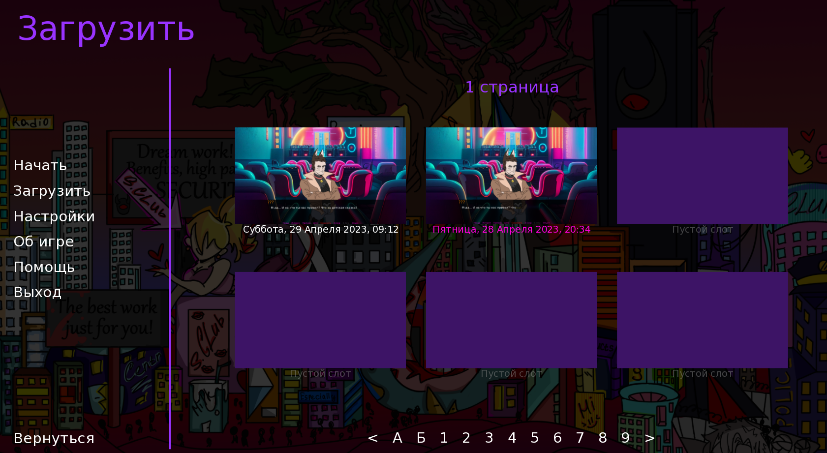


Рис.17 «Экран с сохранениями»

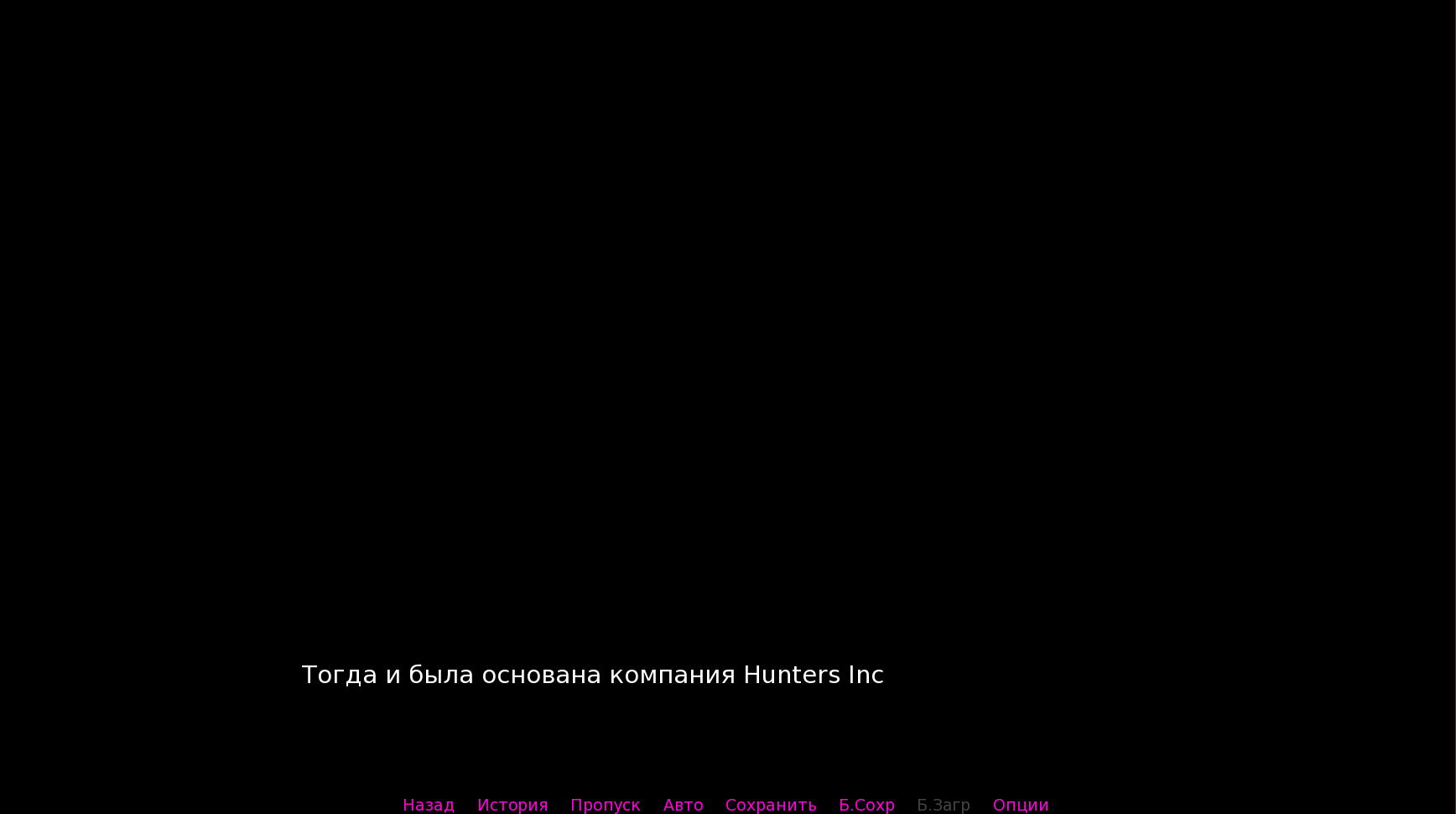


Рис.18 «Начало игры»

5. Тестирование программного модуля

Тестирование программного обеспечения — процесс исследования, испытания программного продукта, имеющий своей целью проверку соответствия между реальным поведением программы и её ожидаемым поведением на конечном наборе тестов, выбранных определённым образом (ISO/IEC TR 19759:2005).

|  |  |
| --- | --- |
| Тестовый пример # | ТС\_ПИ\_1 |
| Приоритет тестирования | Высокий |
| Сохранение | После того как нажимаем на том или ином этапе сохранить, оно должно записаться в выбранный нами слот |
| Краткое изложение теста | Это тестирование проверяет возможность сохранения того или иного этапа игры |
| Этапы теста | Выбрать сохранить 🡪 выбрать слот под сохранение 🡪 сохранить в выбранный слот |
| Тестовые данные | Сохранить |
| Ожидаемый результат | Сохранение запишется |
| Фактический результат | Сохранение записалось |
| Статус | Зачет |

|  |  |
| --- | --- |
| Тестовый пример # | ТС\_ПИ\_2 |
| Приоритет тестирования | Высокий |
| Заголовок/название теста | Изменить параметры экрана с оконного на полноэкранный |
| Краткое изложение теста | Это тестирование проверяет работоспособность изменения экрана |
| Этапы теста | Зайти в настройки 🡪 выбрать полноэкранный режим |
| Тестовые данные | Настройки |
| Ожидаемый результат | Экран перешел в полноэкран |
| Фактический результат | Экран перешел в полноэкран |
| Статус | Зачет |

6. Заключение

В заключение хочу сказать, что представленное программное приложение способно успешно осуществлять выполнение поставленных перед ним задач, однако его функциональные возможности могут быть расширены:

* Сделать игру дольше
* Добавить персонажа, за которого можно было бы играть самому
* Добавить взаимодействие с операционной системой

В процессе разработки данного программного приложения, я углубила свои знания и навыки в области информационных технологий, изучив множество новых и полезных методик и технологий, которые я успешно применила в ходе работы

В процессе разработки были использованы такие технологии как:

1. Python - это высокоуровневый язык программирования, который используется для разработки различных приложений, включая веб-сайты, научные вычисления, игры и многое другое. Python отличается простотой и понятностью синтаксиса, что делает его доступным для новичков в программировании. Он также предлагает множество библиотек и фреймворков, которые упрощают разработку приложений и повышают их производительность. Python является одним из самых популярных языков программирования в мире и используется многими крупными компаниями, такими как Google, Facebook, Dropbox и др.
2. Ren'Py - это свободная и бесплатная среда разработки визуальных новелл, созданная на языке программирования Python. Она предоставляет разработчикам множество инструментов для создания интерактивных книг и игр, включая возможность добавления анимации, звуков и музыки. Ren'Py также позволяет легко создавать персонажей и задавать им диалоги, а также создавать ветвящиеся сюжеты с различными концовками в зависимости от выборов игрока. Благодаря своей простоте и гибкости, Ren'Py стал одним из самых популярных инструментов для создания визуальных новелл.
3. SAI 2 - это программа для создания цифрового иллюстрационного и графического контента, которая предоставляет широкий спектр инструментов для рисования, цветовой коррекции, анимации и многое другое. Она позволяет художникам и дизайнерам создавать профессиональные работы с высоким качеством, используя различные техники и стили. SAI 2 имеет простой и интуитивно понятный интерфейс, что делает его доступным для пользователей всех уровней опыта. Она также поддерживает работу с различными форматами файлов, включая PSD, BMP, JPEG и PNG. SAI 2 используется в основном художниками-иллюстраторами, комиксистами и аниматорами.

7. Список литературы:

1. **«Изучаем Python»,** **Марк Лутц**, «Диалектика», 2019 год.
2. **«Простой Python», Билл Любанович,** «Питер», 2021 год.
3. **«Грокаем глубокое обучение»,** **Эндрю Траск,** «Питер», 2019 год.
4. Документацию Ren Py

# 8. Приложение

Скипт игры

# Вы можете расположить сценарий своей игры в этом файле.

# Определение персонажей игры.

define d = Character('???', color="#ff8800")

define h = Character('Хэкон', color="#a200ffd0")

define D = Character('Дерри', color="#ff8800")

define audio.musnorm = "music/kino.ogg"

define audio.uli = "music/ulica.ogg"

define audio.zlo = "sounds/rrr.ogg"

# Вместо использования оператора image можете просто

# складывать все ваши файлы изображений в папку images.

# Например, сцену bg room можно вызвать файлом "bg room.png",

# а eileen happy — "eileen happy.webp", и тогда они появятся в игре.

# Игра начинается здесь:

init:

$ left2 = Position(xalign=0.1)

$ right2 = Position(xalign=0.6)

#image Lianda norm ="images/Lianda/Lianda norm.png"

label start:

scene room

play music musnorm

#show der norm

"...Далекий 2025 год"

"...год, когда, для человека изменился привычный ему мир"

"Мир стал лучше, тот самый мир технологий-будущее"

"\*ПОМЕХИ\*"

"...именно в тот момент появились ОНИ..."

"Тогда и была основана компания Hunters Inc"

"Именно благодаря Hunters Inc, человечевство еще сущевствует"

scene kino

with fade

scene kino

show der ned

d "М-да... И на что ты нас привел? Что за детская сказка?"

d "Ну ты бы еще на каких нибудь феечек ВИНКС нас отвел блин"

show der zad

d "ХОТЯ НЕЕЕТ!"

#d "Тогда уж давай на каких-нибудь покемонов лучше!"

menu:

"Тогда уж давай на каких-нибудь покемонов лучше!"

"Ну пойдем":

jump pokemon

return

label pokemon:

show der z

play sound zlo

d "НЕ НУ ТЫ СОВСЕМ ЧТО ЛИ?"

d "Это был сарказм! Я пошутил!"

show der ned

d "Совсем уже... Не надо мне еще такого счастья"

d "Так то на дворе 2123 год!"

d "А ты Хэк, привел меня на это старье..."

hide der ned

show hec norm at right

with fade

h "Между прочим, мы на задании, мы сюда не развлекаться пришли"

h "А что насчет фильма, он исторический!"

show hec zlo

h "Думаю, тебе Дерри было бы полезно узнать откуда ты взялся"

show der zad at left

with fade

D "Эй! Я и так знаю откуда я родом, не для нас это все делали, а для вас, людей!"

show der zad

D "Дабы знали, кто может ждать в темной улочке, или что"

show hec norm

h "Ладно, не будем об этом. Лучше давай о задании"

show der norm

D "Да, тот заказ, я с таким скучным фильмом, забыл о нем..."

show der zad

D "Так что там у нас по нему? Есть какая зацепка?"

show hec zlo

h "Нет, потому мы здесь. Этот заказ не только у нас, а так же и у той самой крупной компании из фильма"

stop music fadeout 1

scene ulica

with fade

play music uli

show hec norm at right

h "Компания Hunters Inc, славится не только своей прекрасной работой, сотрудниками и много еще чем"

h "Ну и огромным выбором оружия"

h "Оружие не из простых, а специальное, да и у каждого сотрудника оно индивидуальное..."

show der norm at left

with fade

D "Прям как мое?"

hide der norm

hide hec norm

"Дерри достал два револьвера, было видно что эта чуть ли не увелирная работа!"

"Они были восхетительными, эта работа, сразу видно руки мастера"

"Однако Хэкона они смутили, что-то они ему напоминали..."

show hec norm at right

with fade

h "Наверное... Я не уверен. Откуда они у тебя вообще? Я у тебя их еще не видел"

show der zad at left

with fade

D "А, да так... Помнишь мы ходили на ярмарку? Так вот, они там лежали..."

show der cool

D "Ну я их и взял, что добру пропадать?"

D "Правда они суперские?"

show hec zlo

h "...Взял говоришь... Думаю их скоро пустят в дело..."

D "Думаешь мы опробуем их на задании? Ха-ха!"

"Вдруг до Дерри дошло, о чем ему говорил друг. Медленно повернувшись, и посмотрев что у него за спиной, он впал в ужас"

show der plac

D "Hola amigo!"

hide der plac

hide hec zlo

"Позади друзей стояли недовольные продовцы с ярмарки, а так же какие-то разбойники. И пришли они сюда явно не с хорошими намерениями"

"Хэкон схватил напарника за воротник куртки и умчался с ним подальше от этой толпы"

"Схватить друга была не проблема, его рост составлял 1,45, но не смотря на свой рост, он был крепок и довольно ловок"

Опции игры

## Данный файл содержит настройки, способные изменить вашу игру.

##

## Строки, начинающиеся с двух '#' — комментарии, и вы не должны их

## раскомментировать. Строки, начинающиеся с одной '#' — комментированный код,

## который вы можете раскомментировать, если посчитаете это нужным.

## Основное ####################################################################

## Читаемое название игры. Используется при установке стандартного заголовка

## окна, показывается в интерфейсе и отчётах об ошибках.

##

## Символы "\_()", окружающие название, отмечают его как пригодное для перевода.

define config.name = \_("Novela")

## Определяет, показывать ли заголовок, данный выше, на экране главного меню.

## Установите на False, чтобы спрятать заголовок.

define gui.show\_name = False

## Версия игры.

define config.version = "1.0"

## Текст, помещённый в экран "Об игре". Поместите текст между тройными скобками.

## Для отделения абзацев оставляйте между ними пустую строку.

define gui.about = \_p("""

""")

## Короткое название игры, используемое для исполняемых файлов и директорий при

## постройке дистрибутивов. Оно должно содержать текст формата ASCII и не должно

## содержать пробелы, двоеточия и точки с запятой.

define build.name = "Novela"

## Звуки и музыка ##############################################################

## Эти три переменные управляют, среди прочего, тем, какие микшеры показываются

## игроку по умолчанию. Установка одной из них в False скроет соответствующий

## микшер.

define config.has\_sound = True

define config.has\_music = True

define config.has\_voice = True

## Чтобы разрешить игроку тестировать громкость на звуковом или голосовом

## каналах, раскомментируйте строчку и настройте пример звука для прослушивания.

# define config.sample\_sound = "sample-sound.ogg"

# define config.sample\_voice = "sample-voice.ogg"

## Раскомментируйте следующую строчку, чтобы настроить аудиофайл, который будет

## проигрываться в главном меню. Этот файл продолжит проигрываться во время

## игры, если не будет остановлен, или не начнёт проигрываться другой аудиофайл.

define config.main\_menu\_music = "main-menu-theme.ogg"

## Переходы ####################################################################

##

## Эти переменные задают переходы, используемые в различных событиях. Каждая

## переменная должна задавать переход или None, чтобы указать на то, что переход

## не должен использоваться.

## Вход и выход в игровое меню.

define config.enter\_transition = dissolve

define config.exit\_transition = dissolve

## Переход между экранами игрового меню.

define config.intra\_transition = dissolve

## Переход, используемый после загрузки слота сохранения.

define config.after\_load\_transition = None

## Используется при входе в главное меню после того, как игра закончится.

define config.end\_game\_transition = None

## Переменная, устанавливающая переход, когда старт игры не существует. Вместо

## неё используйте функцию with после показа начальной сценки.

## Управление окнами ###########################################################

##

## Эта строка контролирует, когда появляется диалоговое окно. Если "show" — оно

## всегда показано. Если "hide" — оно показывается, только когда представлен

## диалог. Если "auto" — окно скрыто до появления оператора scene и показывается

## при появлении диалога.

##

## После начала игры этот параметр можно изменить с помощью "window show",

## "window hide" и "window auto".

define config.window = "auto"

## Переходы, используемые при показе и скрытии диалогового окна

define config.window\_show\_transition = Dissolve(.2)

define config.window\_hide\_transition = Dissolve(.2)

## Стандартные настройки #######################################################

## Контролирует стандартную скорость текста. По умолчанию, это 0 — мгновенно,

## в то время как любая другая цифра — это количество символов, печатаемых в

## секунду.

default preferences.text\_cps = 0

## Стандартная задержка авточтения. Большие значения означают долгие ожидания, а

## от 0 до 30 — вполне допустимый диапазон.

default preferences.afm\_time = 15

## Директория сохранений #######################################################

##

## Контролирует зависимое от платформы место, куда Ren'Py будет складывать файлы

## сохранения этой игры. Файлы сохранений будут храниться в:

##

## Windows: %APPDATA\RenPy\<config.save\_directory>

##

## Macintosh: $HOME/Library/RenPy/<config.save\_directory>

##

## Linux: $HOME/.renpy/<config.save\_directory>

##

## Этот параметр обычно не должен изменяться, а если и изменился, должен быть

## текстовой строчкой, а не выражением.

define config.save\_directory = "Novela-1674496280"

## Иконка ######################################################################

##

## Иконка, показываемая на панели задач или на dock.

define config.window\_icon = "gui/window\_icon.png"

## Настройка Дистрибутива ######################################################

##

## Эта секция контролирует, как Ren'Py строит дистрибутивные файлы из вашего

## проекта.

init python:

## Следующие функции берут образцы файлов. Образцы файлов не учитывают

## регистр и соответствующе зависят от директории проекта (base), с или без

## учёта /, задающей директорию. Если обнаруживается множество одноимённых

## файлов, то используется только первый.

##

## Инструкция:

##

## / — разделитель директорий.

##

## \* включает в себя все символы, исключая разделитель директорий.

##

## \*\* включает в себя все символы, включая разделитель директорий.

##

## Например, "\*.txt" берёт все файлы формата txt из директории base, "game/

## \*\*.ogg" берёт все файлы ogg из директории game и всех поддиректорий, а

## "\*\*.psd" берёт все файлы psd из любого места проекта.

## Классифицируйте файлы как None, чтобы исключить их из дистрибутивов.

build.classify('\*\*~', None)

build.classify('\*\*.bak', None)

build.classify('\*\*/.\*\*', None)

build.classify('\*\*/#\*\*', None)

build.classify('\*\*/thumbs.db', None)

## Чтобы архивировать файлы, классифицируйте их, например, как 'archive'.

# build.classify('game/\*\*.png', 'archive')

# build.classify('game/\*\*.jpg', 'archive')

## Файлы, соответствующие образцам документации, дублируются в приложениях

## Mac, чтобы они появлялись и в приложении, и в zip архиве.

build.documentation('\*.html')

build.documentation('\*.txt')

## Лицензионный ключ Google Play требуется для загрузки файлов расширений и

## поддержки внутриигровых покупок. Он может быть найден на странице "Services &

## APIs" консоли разработчика Google Play.

# define build.google\_play\_key = "..."

## Имя пользователя и название проекта, ассоциированные с проектом на itch.io,

## разделённые дробью.

# define build.itch\_project = "renpytom/test-project"

Интерфейс игры

################################################################################

## Инициализация

################################################################################

## Оператор init offset повышает приоритет инициализации в этом файле над

## другими файлами, из-за чего инициализация здесь запускается первее.

init offset = -2

## Вызываю gui.init, чтобы сбросить стили, чувствительные к стандартным

## значениям, и задать высоту и ширину окна игры.

init python:

gui.init(1920, 1080)

################################################################################

## Конфигурируемые Переменные GUI

################################################################################

## Цвета #######################################################################

##

## Цвета текста в интерфейсе.

## Акцентный цвет используется в заголовках и подчёркнутых текстах.

define gui.accent\_color = '#9933ff'

## Цвет, используемый в текстовой кнопке, когда она не выбрана и не наведена.

define gui.idle\_color = '#ffffffff'

## Small\_color используется в маленьком тексте, который должен быть ярче/темнее,

## для того, чтобы выделяться.

define gui.idle\_small\_color = '#ff00dd'

## Цвет, используемых в кнопках и панелях, когда они наведены.

define gui.hover\_color = '#ff0000'

## Цвет, используемый текстовой кнопкой, когда она выбрана, но не наведена.

## Кнопка может быть выбрана, если это текущий экран или текущее значение

## настройки.

define gui.selected\_color = '#ffffff'

## Цвет, используемый текстовой кнопкой, когда она не может быть выбрана.

define gui.insensitive\_color = '#8888887f'

## Цвета, используемые для частей панелей, которые не заполняются. Они

## используются не напрямую, а только при воссоздании файлов изображений.

define gui.muted\_color = '#3d1466'

define gui.hover\_muted\_color = '#5b1e99'

## Цвета, используемые в тексте диалогов и выборов.

define gui.text\_color = '#ffffff'

define gui.interface\_text\_color = '#ffffff'

## Шрифты и их размеры #########################################################

## Шрифт, используемый внутриигровым текстом.

define gui.text\_font = "DejaVuSans.ttf"

## Шрифт, используемый именами персонажей.

define gui.name\_text\_font = "DejaVuSans.ttf"

## Шрифт, используемый текстом вне игры.

define gui.interface\_text\_font = "DejaVuSans.ttf"

## Размер нормального текста диалога.

define gui.text\_size = 33

## Размер имён персонажей.

define gui.name\_text\_size = 45

## Размер текста в пользовательском интерфейсе.

define gui.interface\_text\_size = 33

## Размер заголовков в пользовательском интерфейсе.

define gui.label\_text\_size = 36

## Размер текста на экране уведомлений.

define gui.notify\_text\_size = 24

## Размер заголовка игры.

define gui.title\_text\_size = 75

## Главное и игровое меню. #####################################################

## Изображения, используемые в главном и игровом меню.

define gui.main\_menu\_background = "gui/main\_menu.png"

define gui.game\_menu\_background = "gui/game\_menu.png"

## Диалог ######################################################################

##

## Эти переменные контролируют, как диалог появляется на отдельной строчке.

## Высота текстового окна, содержащего диалог.

define gui.textbox\_height = 278

## Местоположение текстового окна по вертикали экрана. 0.0 — верх, 0.5 — центр и

## 1.0 — низ.

define gui.textbox\_yalign = 1.0

## Местоположение имени говорящего персонажа по отношению к текстовому окну.

## Это могут быть целые значения в пикселях слева и сверху от начала окна или

## процентное отношение, например, 0.5 для центрирования.

define gui.name\_xpos = 360

define gui.name\_ypos = 0

## Горизонтальное выравнивание имени персонажа. Это может быть 0.0 для

## левоориентированного, 0.5 для центрированного и 1.0 для правоориентированного

## выравнивания.

define gui.name\_xalign = 0.0

## Ширина, высота и границы окна, содержащего имя персонажа или None, для

## автоматической размерки.

define gui.namebox\_width = None

define gui.namebox\_height = None

## Границы окна, содержащего имя персонажа слева, сверху, справа и снизу по

## порядку.

define gui.namebox\_borders = Borders(5, 5, 5, 5)

## Если True, фон текстового окна будет моститься (расширяться по эффекту

## плитки). Если False, фон текстового окна будет фиксированным.

define gui.namebox\_tile = False

## Размещение диалога по отношению к текстовому окну. Это могут быть целые

## значения в пикселях слева и сверху от текстового окна или процентное

## отношение, например, 0.5 для центрирования.

define gui.dialogue\_xpos = 402

define gui.dialogue\_ypos = 75

## Максимальная ширина текста диалога в пикселях.

define gui.dialogue\_width = 1116

## Горизонтальное выравнивание текста диалога. Это может быть 0.0 для

## левоориентированного, 0.5 для центрированного и 1.0 для правоориентированного

## выравнивания.

define gui.dialogue\_text\_xalign = 0.0

## Кнопки ######################################################################

##

## Эти переменные, вместе с файлами изображений в gui/button, контролируют

## аспекты того, как отображаются кнопки.

## Ширина и высота кнопки в пикселях. Если None, Ren'Py самостоятельно

## рассчитает размер.

define gui.button\_width = None

define gui.button\_height = None

## Границы каждой стороны кнопки в порядке слева, сверху, справа, снизу.

define gui.button\_borders = Borders(6, 6, 6, 6)

## Если True, фон изображения будет моститься. Если False, фон изображения будет

## линейно масштабирован.

define gui.button\_tile = False

## Шрифт, используемый кнопкой.

define gui.button\_text\_font = gui.interface\_text\_font

## Размер текста, используемый кнопкой.

define gui.button\_text\_size = gui.interface\_text\_size

## Цвет текста в кнопке в различных состояниях.

define gui.button\_text\_idle\_color = gui.idle\_color

define gui.button\_text\_hover\_color = gui.hover\_color

define gui.button\_text\_selected\_color = gui.selected\_color

define gui.button\_text\_insensitive\_color = gui.insensitive\_color

## Горизонтальное выравнивание текста в кнопке. (0.0 — лево, 0.5 — по центру,

## 1.0 — право).

define gui.button\_text\_xalign = 0.0

## Эти переменные переписывают настройки различных видов кнопок. Пожалуйста,

## посмотрите документацию по gui для просмотра всех вариаций кнопок и для чего

## каждая из них нужна.

##

## Эти настройки используются стандартным интерфейсом:

define gui.radio\_button\_borders = Borders(27, 6, 6, 6)

define gui.check\_button\_borders = Borders(27, 6, 6, 6)

define gui.confirm\_button\_text\_xalign = 0.5

define gui.page\_button\_borders = Borders(15, 6, 15, 6)

define gui.quick\_button\_borders = Borders(15, 6, 15, 0)

define gui.quick\_button\_text\_size = 21

define gui.quick\_button\_text\_idle\_color = gui.idle\_small\_color

define gui.quick\_button\_text\_selected\_color = gui.accent\_color

## Вы также можете добавить собственные настройки, добавляя правильно

## именованные переменные. Например, вы можете раскомментировать следующую

## строчку, чтобы установить ширину кнопок навигации.

# define gui.navigation\_button\_width = 250

## Кнопки Выбора ###############################################################

##

## Кнопки выбора используются во внутриигровых меню.

define gui.choice\_button\_width = 1185

define gui.choice\_button\_height = None

define gui.choice\_button\_tile = False

define gui.choice\_button\_borders = Borders(150, 8, 150, 8)

define gui.choice\_button\_text\_font = gui.text\_font

define gui.choice\_button\_text\_size = gui.text\_size

define gui.choice\_button\_text\_xalign = 0.5

define gui.choice\_button\_text\_idle\_color = "#cccccc"

define gui.choice\_button\_text\_hover\_color = "#ffffff"

define gui.choice\_button\_text\_insensitive\_color = "#444444"

## Кнопки Слотов ###############################################################

##

## Кнопка слотов — особый вид кнопки. Она содержит миниатюру и текст,

## описывающий слот сохранения. Слот сохранения использует файлы из gui/button,

## как и другие виды кнопок.

## Кнопка слота сохранения.

define gui.slot\_button\_width = 414

define gui.slot\_button\_height = 309

define gui.slot\_button\_borders = Borders(15, 15, 15, 15)

define gui.slot\_button\_text\_size = 21

define gui.slot\_button\_text\_xalign = 0.5

define gui.slot\_button\_text\_idle\_color = gui.idle\_small\_color

define gui.slot\_button\_text\_selected\_idle\_color = gui.selected\_color

define gui.slot\_button\_text\_selected\_hover\_color = gui.hover\_color

## Ширина и высота миниатюры, используемой слотом сохранения.

define config.thumbnail\_width = 384

define config.thumbnail\_height = 216

## Количество колонок и рядов в таблице слотов.

define gui.file\_slot\_cols = 3

define gui.file\_slot\_rows = 2

## Позиционирование и Интервалы ################################################

##

## Эти переменные контролируют позиционирование и интервалы различных элементов

## пользовательского интерфейса.

## Местоположение левого края навигационных кнопок по отношению к левому краю

## экрана.

define gui.navigation\_xpos = 60

## Вертикальная позиция индикатора пропуска.

define gui.skip\_ypos = 15

## Вертикальная позиция экрана уведомлений.

define gui.notify\_ypos = 68

## Интервал между выборами в меню.

define gui.choice\_spacing = 33

## Кнопки в секции навигации главного и игрового меню.

define gui.navigation\_spacing = 6

## Контролирует интервал между настройками.

define gui.pref\_spacing = 15

## Контролирует интервал между кнопками настройки.

define gui.pref\_button\_spacing = 0

## Интервал между кнопками страниц.

define gui.page\_spacing = 0

## Интервал между слотами.

define gui.slot\_spacing = 15

## Позиция текста главного меню.

define gui.main\_menu\_text\_xalign = 1.0

## Рамки #######################################################################

##

## Эти переменные контролируют вид рамок, содержащих компоненты

## пользовательского интерфейса, когда наложения или окна не представлены.

## Генерируем рамки.

define gui.frame\_borders = Borders(6, 6, 6, 6)

## Рамки, используемые в частях экрана подтверждения.

define gui.confirm\_frame\_borders = Borders(60, 60, 60, 60)

## Рамки, используемые в частях экрана пропуска.

define gui.skip\_frame\_borders = Borders(24, 8, 75, 8)

## Рамки, используемые в частях экрана уведомлений.

define gui.notify\_frame\_borders = Borders(24, 8, 60, 8)

## Должны ли фоны рамок моститься?

define gui.frame\_tile = False

## Панели, Полосы прокрутки и Ползунки #########################################

##

## Эти настройки контролируют вид и размер панелей, полос прокрутки и ползунков.

##

## Стандартный GUI использует только ползунки и вертикальные полосы прокрутки.

## Все остальные полосы используются только в новосозданных экранах.

## Высота горизонтальных панелей, полос прокрутки и ползунков. Ширина

## вертикальных панелей, полос прокрутки и ползунков.

define gui.bar\_size = 38

define gui.scrollbar\_size = 18

define gui.slider\_size = 38

## True, если изображения панелей должны моститься. False, если они должны быть

## линейно масштабированы.

define gui.bar\_tile = False

define gui.scrollbar\_tile = False

define gui.slider\_tile = False

## Горизонтальные границы.

define gui.bar\_borders = Borders(6, 6, 6, 6)

define gui.scrollbar\_borders = Borders(6, 6, 6, 6)

define gui.slider\_borders = Borders(6, 6, 6, 6)

## Вертикальные границы.

define gui.vbar\_borders = Borders(6, 6, 6, 6)

define gui.vscrollbar\_borders = Borders(6, 6, 6, 6)

define gui.vslider\_borders = Borders(6, 6, 6, 6)

## Что делать с непрокручиваемыми полосами прокрутки в интерфейсе. "hide" прячет

## их, а None их показывает.

define gui.unscrollable = "hide"

## История #####################################################################

##

## Экран истории показывает диалог, который игрок уже прошёл.

## Количество диалоговых блоков истории, которые Ren'Py будет хранить.

define config.history\_length = 250

## Высота доступных записей на экране истории, или None, чтобы задать высоту в

## зависимости от производительности.

define gui.history\_height = 210

## Местоположение, ширина и выравнивание заголовка, показывающего имя говорящего

## персонажа.

define gui.history\_name\_xpos = 233

define gui.history\_name\_ypos = 0

define gui.history\_name\_width = 233

define gui.history\_name\_xalign = 1.0

## Местоположение, ширина и выравнивание диалогового текста.

define gui.history\_text\_xpos = 255

define gui.history\_text\_ypos = 3

define gui.history\_text\_width = 1110

define gui.history\_text\_xalign = 0.0

## Режим NVL ###################################################################

##

## Экран режима NVL показывает диалог NVL персонажей.

## Границы фона окна NVL.

define gui.nvl\_borders = Borders(0, 15, 0, 30)

## Максимальное число показываемых строк в режиме NVL. Когда количество строчек

## начинает превышать это значение, старые строчки очищаются.

define gui.nvl\_list\_length = 6

## Высота доступных строчек в режиме NVL. Установите на None, чтобы строчки

## динамически регулировали свою высоту.

define gui.nvl\_height = 173

## Интервал между строчками в режиме NVL, если gui.nvl\_height имеет значение

## None, а также между строчками и меню режима NVL.

define gui.nvl\_spacing = 15

## Местоположение, ширина и выравнивание заголовка, показывающего имя говорящего

## персонажа.

define gui.nvl\_name\_xpos = 645

define gui.nvl\_name\_ypos = 0

define gui.nvl\_name\_width = 225

define gui.nvl\_name\_xalign = 1.0

## Местоположение, ширина и выравнивание диалогового текста.

define gui.nvl\_text\_xpos = 675

define gui.nvl\_text\_ypos = 12

define gui.nvl\_text\_width = 885

define gui.nvl\_text\_xalign = 0.0

## Местоположение, ширина и выравнивание текста nvl\_thought (текст от лица

## персонажа nvl\_narrator).

define gui.nvl\_thought\_xpos = 360

define gui.nvl\_thought\_ypos = 0

define gui.nvl\_thought\_width = 1170

define gui.nvl\_thought\_xalign = 0.0

## Местоположение кнопок меню NVL.

define gui.nvl\_button\_xpos = 675

define gui.nvl\_button\_xalign = 0.0

## Локализация #################################################################

## Эта настройка контролирует доступ к разрыву линий. Стандартная настройка

## подходит для большинства языков. Список доступных значений можно найти на

## https://www.renpy.org/doc/html/style\_properties.html#style-property-language

define gui.language = "unicode"

################################################################################

## Мобильные устройства

################################################################################

init python:

## Этот параметр увеличивает размер быстрых кнопок, чтобы сделать их

## доступнее для нажатия на планшетах и телефонах.

@gui.variant

def touch():

gui.quick\_button\_borders = Borders(60, 21, 60, 0)

## Это изменяет размеры и интервалы различных элементов GUI, чтобы

## убедиться, что они будут лучше видны на телефонах.

@gui.variant

def small():

## Размеры шрифтов.

gui.text\_size = 45

gui.name\_text\_size = 54

gui.notify\_text\_size = 38

gui.interface\_text\_size = 45

gui.button\_text\_size = 45

gui.label\_text\_size = 51

## Регулирует местоположение текстового окна.

gui.textbox\_height = 360

gui.name\_xpos = 120

gui.dialogue\_xpos = 135

gui.dialogue\_width = 1650

## Изменяет размеры и интервалы различных объектов.

gui.slider\_size = 54

gui.choice\_button\_width = 1860

gui.choice\_button\_text\_size = 45

gui.navigation\_spacing = 30

gui.pref\_button\_spacing = 15

gui.history\_height = 285

gui.history\_text\_width = 1035

gui.quick\_button\_text\_size = 30

## Местоположение кнопок слотов.

gui.file\_slot\_cols = 2

gui.file\_slot\_rows = 2

## Режим NVL.

gui.nvl\_height = 255

gui.nvl\_name\_width = 458

gui.nvl\_name\_xpos = 488

gui.nvl\_text\_width = 1373

gui.nvl\_text\_xpos = 518

gui.nvl\_text\_ypos = 8

gui.nvl\_thought\_width = 1860

gui.nvl\_thought\_xpos = 30

gui.nvl\_button\_width = 1860

gui.nvl\_button\_xpos = 30

Экран

################################################################################

## Инициализация

################################################################################

init offset = -1

################################################################################

## Стили

################################################################################

style default:

properties gui.text\_properties()

language gui.language

style input:

properties gui.text\_properties("input", accent=True)

adjust\_spacing False

style hyperlink\_text:

properties gui.text\_properties("hyperlink", accent=True)

hover\_underline True

style gui\_text:

properties gui.text\_properties("interface")

style button:

properties gui.button\_properties("button")

style button\_text is gui\_text:

properties gui.text\_properties("button")

yalign 0.5

style label\_text is gui\_text:

properties gui.text\_properties("label", accent=True)

style prompt\_text is gui\_text:

properties gui.text\_properties("prompt")

style bar:

ysize gui.bar\_size

left\_bar Frame("gui/bar/left.png", gui.bar\_borders, tile=gui.bar\_tile)

right\_bar Frame("gui/bar/right.png", gui.bar\_borders, tile=gui.bar\_tile)

style vbar:

xsize gui.bar\_size

top\_bar Frame("gui/bar/top.png", gui.vbar\_borders, tile=gui.bar\_tile)

bottom\_bar Frame("gui/bar/bottom.png", gui.vbar\_borders, tile=gui.bar\_tile)

style scrollbar:

ysize gui.scrollbar\_size

base\_bar Frame("gui/scrollbar/horizontal\_[prefix\_]bar.png", gui.scrollbar\_borders, tile=gui.scrollbar\_tile)

thumb Frame("gui/scrollbar/horizontal\_[prefix\_]thumb.png", gui.scrollbar\_borders, tile=gui.scrollbar\_tile)

style vscrollbar:

xsize gui.scrollbar\_size

base\_bar Frame("gui/scrollbar/vertical\_[prefix\_]bar.png", gui.vscrollbar\_borders, tile=gui.scrollbar\_tile)

thumb Frame("gui/scrollbar/vertical\_[prefix\_]thumb.png", gui.vscrollbar\_borders, tile=gui.scrollbar\_tile)

style slider:

ysize gui.slider\_size

base\_bar Frame("gui/slider/horizontal\_[prefix\_]bar.png", gui.slider\_borders, tile=gui.slider\_tile)

thumb "gui/slider/horizontal\_[prefix\_]thumb.png"

style vslider:

xsize gui.slider\_size

base\_bar Frame("gui/slider/vertical\_[prefix\_]bar.png", gui.vslider\_borders, tile=gui.slider\_tile)

thumb "gui/slider/vertical\_[prefix\_]thumb.png"

style frame:

padding gui.frame\_borders.padding

background Frame("gui/frame.png", gui.frame\_borders, tile=gui.frame\_tile)

################################################################################

## Внутриигровые экраны

################################################################################

## Экран разговора #############################################################

##

## Экран разговора используется для показа диалога игроку. Он использует два

## параметра — who и what — что, соответственно, имя говорящего персонажа и

## показываемый текст. (Параметр who может быть None, если имя не задано.)

##

## Этот экран должен создать текст с id "what", чтобы Ren'Py могла показать

## текст. Здесь также можно создать наложения с id "who" и id "window", чтобы

## применить к ним настройки стиля.

##

## https://www.renpy.org/doc/html/screen\_special.html#say

screen say(who, what):

style\_prefix "say"

window:

id "window"

if who is not None:

window:

id "namebox"

style "namebox"

text who id "who"

text what id "what"

## Если есть боковое изображение ("голова"), показывает её поверх текста.

## По стандарту не показывается на варианте для мобильных устройств — мало

## места.

if not renpy.variant("small"):

add SideImage() xalign 0.0 yalign 1.0

## Делает namebox доступным для стилизации через объект Character.

init python:

config.character\_id\_prefixes.append('namebox')

style window is default

style say\_label is default

style say\_dialogue is default

style say\_thought is say\_dialogue

style namebox is default

style namebox\_label is say\_label

style window:

xalign 0.5

xfill True

yalign gui.textbox\_yalign

ysize gui.textbox\_height

background Image("gui/textbox.png", xalign=0.5, yalign=1.0)

style namebox:

xpos gui.name\_xpos

xanchor gui.name\_xalign

xsize gui.namebox\_width

ypos gui.name\_ypos

ysize gui.namebox\_height

background Frame("gui/namebox.png", gui.namebox\_borders, tile=gui.namebox\_tile, xalign=gui.name\_xalign)

padding gui.namebox\_borders.padding

style say\_label:

properties gui.text\_properties("name", accent=True)

xalign gui.name\_xalign

yalign 0.5

style say\_dialogue:

properties gui.text\_properties("dialogue")

xpos gui.dialogue\_xpos

xsize gui.dialogue\_width

ypos gui.dialogue\_ypos

adjust\_spacing False

## Экран ввода #################################################################

##

## Этот экран используется, чтобы показывать renpy.input. Это параметр запроса,

## используемый для того, чтобы дать игроку ввести в него текст.

##

## Этот экран должен создать наложение ввода с id "input", чтобы принять

## различные вводимые параметры.

##

## https://www.renpy.org/doc/html/screen\_special.html#input

screen input(prompt):

style\_prefix "input"

window:

vbox:

xanchor gui.dialogue\_text\_xalign

xpos gui.dialogue\_xpos

xsize gui.dialogue\_width

ypos gui.dialogue\_ypos

text prompt style "input\_prompt"

input id "input"

style input\_prompt is default

style input\_prompt:

xalign gui.dialogue\_text\_xalign

properties gui.text\_properties("input\_prompt")

style input:

xalign gui.dialogue\_text\_xalign

xmaximum gui.dialogue\_width

## Экран выбора ################################################################

##

## Этот экран используется, чтобы показывать внутриигровые выборы,

## представленные оператором menu. Один параметр, вложения, список объектов,

## каждый с заголовком и полями действия.

##

## https://www.renpy.org/doc/html/screen\_special.html#choice

screen choice(items):

style\_prefix "choice"

vbox:

for i in items:

textbutton i.caption action i.action

style choice\_vbox is vbox

style choice\_button is button

style choice\_button\_text is button\_text

style choice\_vbox:

xalign 0.5

ypos 405

yanchor 0.5

spacing gui.choice\_spacing

style choice\_button is default:

properties gui.button\_properties("choice\_button")

style choice\_button\_text is default:

properties gui.button\_text\_properties("choice\_button")

## Экран быстрого меню #########################################################

##

## Быстрое меню показывается внутри игры, чтобы обеспечить лёгкий доступ к

## внеигровым меню.

screen quick\_menu():

## Гарантирует, что оно появляется поверх других экранов.

zorder 100

if quick\_menu:

hbox:

style\_prefix "quick"

xalign 0.5

yalign 1.0

textbutton \_("Назад") action Rollback()

textbutton \_("История") action ShowMenu('history')

textbutton \_("Пропуск") action Skip() alternate Skip(fast=True, confirm=True)

textbutton \_("Авто") action Preference("auto-forward", "toggle")

textbutton \_("Сохранить") action ShowMenu('save')

textbutton \_("Б.Сохр") action QuickSave()

textbutton \_("Б.Загр") action QuickLoad()

textbutton \_("Опции") action ShowMenu('preferences')

## Данный код гарантирует, что экран быстрого меню будет показан в игре в любое

## время, если только игрок не скроет интерфейс.

init python:

config.overlay\_screens.append("quick\_menu")

default quick\_menu = True

style quick\_button is default

style quick\_button\_text is button\_text

style quick\_button:

properties gui.button\_properties("quick\_button")

style quick\_button\_text:

properties gui.button\_text\_properties("quick\_button")

################################################################################

## Экраны Главного и Игрового меню

################################################################################

## Экран навигации #############################################################

##

## Этот экран включает в себя главное и игровое меню, и обеспечивает навигацию к

## другим меню и к началу игры.

screen navigation():

vbox:

style\_prefix "navigation"

xpos gui.navigation\_xpos

yalign 0.5

spacing gui.navigation\_spacing

if main\_menu:

textbutton \_("Начать") action Start()

else:

textbutton \_("История") action ShowMenu("history")

textbutton \_("Сохранить") action ShowMenu("save")

textbutton \_("Загрузить") action ShowMenu("load")

textbutton \_("Настройки") action ShowMenu("preferences")

if \_in\_replay:

textbutton \_("Завершить повтор") action EndReplay(confirm=True)

elif not main\_menu:

textbutton \_("Главное меню") action MainMenu()

textbutton \_("Об игре") action ShowMenu("about")

if renpy.variant("pc") or (renpy.variant("web") and not renpy.variant("mobile")):

## Помощь не необходима и не относится к мобильным устройствам.

textbutton \_("Помощь") action ShowMenu("help")

if renpy.variant("pc"):

## Кнопка выхода блокирована в iOS и не нужна на Android и в веб-

## версии.

textbutton \_("Выход") action Quit(confirm=not main\_menu)

style navigation\_button is gui\_button

style navigation\_button\_text is gui\_button\_text

style navigation\_button:

size\_group "navigation"

properties gui.button\_properties("navigation\_button")

style navigation\_button\_text:

properties gui.button\_text\_properties("navigation\_button")

## Экран главного меню #########################################################

##

## Используется, чтобы показать главное меню после запуска игры.

##

## https://www.renpy.org/doc/html/screen\_special.html#main-menu

screen main\_menu():

## Этот тег гарантирует, что любой другой экран с тем же тегом будет

## заменять этот.

tag menu

add gui.main\_menu\_background

## Эта пустая рамка затеняет главное меню.

frame:

style "main\_menu\_frame"

## Оператор use включает отображение другого экрана в данном. Актуальное

## содержание главного меню находится на экране навигации.

use navigation

if gui.show\_name:

vbox:

style "main\_menu\_vbox"

text "[config.name!t]":

style "main\_menu\_title"

text "[config.version]":

style "main\_menu\_version"

style main\_menu\_frame is empty

style main\_menu\_vbox is vbox

style main\_menu\_text is gui\_text

style main\_menu\_title is main\_menu\_text

style main\_menu\_version is main\_menu\_text

style main\_menu\_frame:

xsize 420

yfill True

background "gui/overlay/main\_menu.png"

style main\_menu\_vbox:

xalign 1.0

xoffset -30

xmaximum 1200

yalign 1.0

yoffset -30

style main\_menu\_text:

properties gui.text\_properties("main\_menu", accent=True)

style main\_menu\_title:

properties gui.text\_properties("title")

style main\_menu\_version:

properties gui.text\_properties("version")

## Экран игрового меню #########################################################

##

## Всё это показывает основную, обобщённую структуру экрана игрового меню. Он

## вызывается с экраном заголовка и показывает фон, заголовок и навигацию.

##

## Параметр scroll может быть None, или "viewport", или "vpgrid", когда этот

## экран предназначается для использования с более чем одним дочерним экраном,

## включённым в него.

screen game\_menu(title, scroll=None, yinitial=0.0):

style\_prefix "game\_menu"

if main\_menu:

add gui.main\_menu\_background

else:

add gui.game\_menu\_background

frame:

style "game\_menu\_outer\_frame"

hbox:

## Резервирует пространство для навигации.

frame:

style "game\_menu\_navigation\_frame"

frame:

style "game\_menu\_content\_frame"

if scroll == "viewport":

viewport:

yinitial yinitial

scrollbars "vertical"

mousewheel True

draggable True

pagekeys True

side\_yfill True

vbox:

transclude

elif scroll == "vpgrid":

vpgrid:

cols 1

yinitial yinitial

scrollbars "vertical"

mousewheel True

draggable True

pagekeys True

side\_yfill True

transclude

else:

transclude

use navigation

textbutton \_("Вернуться"):

style "return\_button"

action Return()

label title

if main\_menu:

key "game\_menu" action ShowMenu("main\_menu")

style game\_menu\_outer\_frame is empty

style game\_menu\_navigation\_frame is empty

style game\_menu\_content\_frame is empty

style game\_menu\_viewport is gui\_viewport

style game\_menu\_side is gui\_side

style game\_menu\_scrollbar is gui\_vscrollbar

style game\_menu\_label is gui\_label

style game\_menu\_label\_text is gui\_label\_text

style return\_button is navigation\_button

style return\_button\_text is navigation\_button\_text

style game\_menu\_outer\_frame:

bottom\_padding 45

top\_padding 180

background "gui/overlay/game\_menu.png"

style game\_menu\_navigation\_frame:

xsize 420

yfill True

style game\_menu\_content\_frame:

left\_margin 60

right\_margin 30

top\_margin 15

style game\_menu\_viewport:

xsize 1380

style game\_menu\_vscrollbar:

unscrollable gui.unscrollable

style game\_menu\_side:

spacing 15

style game\_menu\_label:

xpos 75

ysize 180

style game\_menu\_label\_text:

size gui.title\_text\_size

color gui.accent\_color

yalign 0.5

style return\_button:

xpos gui.navigation\_xpos

yalign 1.0

yoffset -45

## Экран Об игре ###############################################################

##

## Этот экран показывает авторскую информацию об игре и Ren'Py.

##

## В этом экране нет ничего особенного, и он служит только примером того, каким

## можно сделать свой экран.

screen about():

tag menu

## Этот оператор включает игровое меню внутрь этого экрана. Дочерний vbox

## включён в порт просмотра внутри экрана игрового меню.

use game\_menu(\_("Об игре"), scroll="viewport"):

style\_prefix "about"

vbox:

label "[config.name!t]"

text \_("Версия [config.version!t]\n")

## gui.about обычно установлено в options.rpy.

if gui.about:

text "[gui.about!t]\n"

text \_("Сделано с помощью {a=https://www.renpy.org/}Ren'Py{/a} [renpy.version\_only].\n\n[renpy.license!t]")

style about\_label is gui\_label

style about\_label\_text is gui\_label\_text

style about\_text is gui\_text

style about\_label\_text:

size gui.label\_text\_size

## Экраны загрузки и сохранения ################################################

##

## Эти экраны ответственны за возможность сохранять и загружать игру. Так

## как они почти одинаковые, оба реализованы по правилам третьего экрана —

## file\_slots.

##

## https://www.renpy.org/doc/html/screen\_special.html#save

screen save():

tag menu

use file\_slots(\_("Сохранить"))

screen load():

tag menu

use file\_slots(\_("Загрузить"))

screen file\_slots(title):

default page\_name\_value = FilePageNameInputValue(pattern=\_("{} страница"), auto=\_("Автосохранения"), quick=\_("Быстрые сохранения"))

use game\_menu(title):

fixed:

## Это гарантирует, что ввод будет принимать enter перед остальными

## кнопками.

order\_reverse True

## Номер страницы, который может быть изменён посредством клика на

## кнопку.

button:

style "page\_label"

key\_events True

xalign 0.5

action page\_name\_value.Toggle()

input:

style "page\_label\_text"

value page\_name\_value

## Таблица слотов.

grid gui.file\_slot\_cols gui.file\_slot\_rows:

style\_prefix "slot"

xalign 0.5

yalign 0.5

spacing gui.slot\_spacing

for i in range(gui.file\_slot\_cols \* gui.file\_slot\_rows):

$ slot = i + 1

button:

action FileAction(slot)

has vbox

add FileScreenshot(slot) xalign 0.5

text FileTime(slot, format=\_("{#file\_time}%A, %d %B %Y, %H:%M"), empty=\_("Пустой слот")):

style "slot\_time\_text"

text FileSaveName(slot):

style "slot\_name\_text"

key "save\_delete" action FileDelete(slot)

## Кнопки для доступа к другим страницам.

hbox:

style\_prefix "page"

xalign 0.5

yalign 1.0

spacing gui.page\_spacing

textbutton \_("<") action FilePagePrevious()

if config.has\_autosave:

textbutton \_("{#auto\_page}А") action FilePage("auto")

if config.has\_quicksave:

textbutton \_("{#quick\_page}Б") action FilePage("quick")

## range(1, 10) задаёт диапазон значений от 1 до 9.

for page in range(1, 10):

textbutton "[page]" action FilePage(page)

textbutton \_(">") action FilePageNext()

style page\_label is gui\_label

style page\_label\_text is gui\_label\_text

style page\_button is gui\_button

style page\_button\_text is gui\_button\_text

style slot\_button is gui\_button

style slot\_button\_text is gui\_button\_text

style slot\_time\_text is slot\_button\_text

style slot\_name\_text is slot\_button\_text

style page\_label:

xpadding 75

ypadding 5

style page\_label\_text:

text\_align 0.5

layout "subtitle"

hover\_color gui.hover\_color

style page\_button:

properties gui.button\_properties("page\_button")

style page\_button\_text:

properties gui.button\_text\_properties("page\_button")

style slot\_button:

properties gui.button\_properties("slot\_button")

style slot\_button\_text:

properties gui.button\_text\_properties("slot\_button")

## Экран настроек ##############################################################

##

## Экран настроек позволяет игроку настраивать игру под себя.

##

## https://www.renpy.org/doc/html/screen\_special.html#preferences

screen preferences():

tag menu

use game\_menu(\_("Настройки"), scroll="viewport"):

vbox:

hbox:

box\_wrap True

if renpy.variant("pc") or renpy.variant("web"):

vbox:

style\_prefix "radio"

label \_("Режим экрана")

textbutton \_("Оконный") action Preference("display", "window")

textbutton \_("Полный") action Preference("display", "fullscreen")

vbox:

style\_prefix "check"

label \_("Пропуск")

textbutton \_("Всего текста") action Preference("skip", "toggle")

textbutton \_("После выборов") action Preference("after choices", "toggle")

textbutton \_("Переходов") action InvertSelected(Preference("transitions", "toggle"))

## Дополнительные vbox'ы типа "radio\_pref" или "check\_pref"

## могут быть добавлены сюда для добавления новых настроек.

null height (4 \* gui.pref\_spacing)

hbox:

style\_prefix "slider"

box\_wrap True

vbox:

label \_("Скорость текста")

bar value Preference("text speed")

label \_("Скорость авточтения")

bar value Preference("auto-forward time")

vbox:

if config.has\_music:

label \_("Громкость музыки")

hbox:

bar value Preference("music volume")

if config.has\_sound:

label \_("Громкость звуков")

hbox:

bar value Preference("sound volume")

if config.sample\_sound:

textbutton \_("Тест") action Play("sound", config.sample\_sound)

if config.has\_voice:

label \_("Громкость голоса")

hbox:

bar value Preference("voice volume")

if config.sample\_voice:

textbutton \_("Тест") action Play("voice", config.sample\_voice)

if config.has\_music or config.has\_sound or config.has\_voice:

null height gui.pref\_spacing

textbutton \_("Без звука"):

action Preference("all mute", "toggle")

style "mute\_all\_button"

style pref\_label is gui\_label

style pref\_label\_text is gui\_label\_text

style pref\_vbox is vbox

style radio\_label is pref\_label

style radio\_label\_text is pref\_label\_text

style radio\_button is gui\_button

style radio\_button\_text is gui\_button\_text

style radio\_vbox is pref\_vbox

style check\_label is pref\_label

style check\_label\_text is pref\_label\_text

style check\_button is gui\_button

style check\_button\_text is gui\_button\_text

style check\_vbox is pref\_vbox

style slider\_label is pref\_label

style slider\_label\_text is pref\_label\_text

style slider\_slider is gui\_slider

style slider\_button is gui\_button

style slider\_button\_text is gui\_button\_text

style slider\_pref\_vbox is pref\_vbox

style mute\_all\_button is check\_button

style mute\_all\_button\_text is check\_button\_text

style pref\_label:

top\_margin gui.pref\_spacing

bottom\_margin 3

style pref\_label\_text:

yalign 1.0

style pref\_vbox:

xsize 338

style radio\_vbox:

spacing gui.pref\_button\_spacing

style radio\_button:

properties gui.button\_properties("radio\_button")

foreground "gui/button/radio\_[prefix\_]foreground.png"

style radio\_button\_text:

properties gui.button\_text\_properties("radio\_button")

style check\_vbox:

spacing gui.pref\_button\_spacing

style check\_button:

properties gui.button\_properties("check\_button")

foreground "gui/button/check\_[prefix\_]foreground.png"

style check\_button\_text:

properties gui.button\_text\_properties("check\_button")

style slider\_slider:

xsize 525

style slider\_button:

properties gui.button\_properties("slider\_button")

yalign 0.5

left\_margin 15

style slider\_button\_text:

properties gui.button\_text\_properties("slider\_button")

style slider\_vbox:

xsize 675

## Экран истории ###############################################################

##

## Этот экран показывает игроку историю диалогов. Хотя в этом экране нет ничего

## особенного, он имеет доступ к истории диалогов, хранимом в \_history\_list.

##

## https://www.renpy.org/doc/html/history.html

screen history():

tag menu

## Избегайте предсказывания этого экрана, так как он может быть очень

## массивным.

predict False

use game\_menu(\_("История"), scroll=("vpgrid" if gui.history\_height else "viewport"), yinitial=1.0):

style\_prefix "history"

for h in \_history\_list:

window:

## Это всё правильно уравняет, если history\_height будет

## установлен на None.

has fixed:

yfit True

if h.who:

label h.who:

style "history\_name"

substitute False

## Берёт цвет из who параметра персонажа, если он

## установлен.

if "color" in h.who\_args:

text\_color h.who\_args["color"]

$ what = renpy.filter\_text\_tags(h.what, allow=gui.history\_allow\_tags)

text what:

substitute False

if not \_history\_list:

label \_("История диалогов пуста.")

## Это определяет, какие теги могут отображаться на экране истории.

define gui.history\_allow\_tags = { "alt", "noalt", "rt", "rb", "art" }

style history\_window is empty

style history\_name is gui\_label

style history\_name\_text is gui\_label\_text

style history\_text is gui\_text

style history\_label is gui\_label

style history\_label\_text is gui\_label\_text

style history\_window:

xfill True

ysize gui.history\_height

style history\_name:

xpos gui.history\_name\_xpos

xanchor gui.history\_name\_xalign

ypos gui.history\_name\_ypos

xsize gui.history\_name\_width

style history\_name\_text:

min\_width gui.history\_name\_width

text\_align gui.history\_name\_xalign

style history\_text:

xpos gui.history\_text\_xpos

ypos gui.history\_text\_ypos

xanchor gui.history\_text\_xalign

xsize gui.history\_text\_width

min\_width gui.history\_text\_width

text\_align gui.history\_text\_xalign

layout ("subtitle" if gui.history\_text\_xalign else "tex")

style history\_label:

xfill True

style history\_label\_text:

xalign 0.5

## Экран помощи ################################################################

##

## Экран, дающий информацию о клавишах управления. Он использует другие экраны

## (keyboard\_help, mouse\_help, и gamepad\_help), чтобы показывать актуальную

## помощь.

screen help():

tag menu

default device = "keyboard"

use game\_menu(\_("Помощь"), scroll="viewport"):

style\_prefix "help"

vbox:

spacing 23

hbox:

textbutton \_("Клавиатура") action SetScreenVariable("device", "keyboard")

textbutton \_("Мышь") action SetScreenVariable("device", "mouse")

if GamepadExists():

textbutton \_("Геймпад") action SetScreenVariable("device", "gamepad")

if device == "keyboard":

use keyboard\_help

elif device == "mouse":

use mouse\_help

elif device == "gamepad":

use gamepad\_help

screen keyboard\_help():

hbox:

label \_("Enter")

text \_("Прохождение диалогов, активация интерфейса.")

hbox:

label \_("Пробел")

text \_("Прохождение диалогов без возможности делать выбор.")

hbox:

label \_("Стрелки")

text \_("Навигация по интерфейсу.")

hbox:

label \_("Esc")

text \_("Вход в игровое меню.")

hbox:

label \_("Ctrl")

text \_("Пропускает диалоги, пока зажат.")

hbox:

label \_("Tab")

text \_("Включает режим пропуска.")

hbox:

label \_("Page Up")

text \_("Откат назад по сюжету игры.")

hbox:

label \_("Page Down")

text \_("Откатывает предыдущее действие вперёд.")

hbox:

label "H"

text \_("Скрывает интерфейс пользователя.")

hbox:

label "S"

text \_("Делает снимок экрана.")

hbox:

label "V"

text \_("Включает поддерживаемый {a=https://www.renpy.org/l/voicing}синтезатор речи{/a}.")

hbox:

label "Shift+A"

text \_("Открывает меню специальных возможностей.")

screen mouse\_help():

hbox:

label \_("Левый клик")

text \_("Прохождение диалогов, активация интерфейса.")

hbox:

label \_("Клик колёсиком")

text \_("Скрывает интерфейс пользователя.")

hbox:

label \_("Правый клик")

text \_("Вход в игровое меню.")

hbox:

label \_("Колёсико вверх\nКлик на сторону отката")

text \_("Откат назад по сюжету игры.")

hbox:

label \_("Колёсико вниз")

text \_("Откатывает предыдущее действие вперёд.")

screen gamepad\_help():

hbox:

label \_("Правый триггер\nA/Нижняя кнопка")

text \_("Прохождение диалогов, активация интерфейса.")

hbox:

label \_("Левый Триггер\nЛевый Бампер")

text \_("Откат назад по сюжету игры.")

hbox:

label \_("Правый бампер")

text \_("Откатывает предыдущее действие вперёд.")

hbox:

label \_("Крестовина, Стики")

text \_("Навигация по интерфейсу.")

hbox:

label \_("Start, Guide")

text \_("Вход в игровое меню.")

hbox:

label \_("Y/Верхняя кнопка")

text \_("Скрывает интерфейс пользователя.")

textbutton \_("Калибровка") action GamepadCalibrate()

style help\_button is gui\_button

style help\_button\_text is gui\_button\_text

style help\_label is gui\_label

style help\_label\_text is gui\_label\_text

style help\_text is gui\_text

style help\_button:

properties gui.button\_properties("help\_button")

xmargin 12

style help\_button\_text:

properties gui.button\_text\_properties("help\_button")

style help\_label:

xsize 375

right\_padding 30

style help\_label\_text:

size gui.text\_size

xalign 1.0

text\_align 1.0

################################################################################

## Дополнительные экраны

################################################################################

## Экран подтверждения #########################################################

##

## Экран подтверждения вызывается, когда Ren'Py хочет спросить у игрока вопрос

## Да или Нет.

##

## https://www.renpy.org/doc/html/screen\_special.html#confirm

screen confirm(message, yes\_action, no\_action):

## Гарантирует, что другие экраны будут недоступны, пока показан этот экран.

modal True

zorder 200

style\_prefix "confirm"

add "gui/overlay/confirm.png"

frame:

vbox:

xalign .5

yalign .5

spacing 45

label \_(message):

style "confirm\_prompt"

xalign 0.5

hbox:

xalign 0.5

spacing 150

textbutton \_("Да") action yes\_action

textbutton \_("Нет") action no\_action

## Правый клик и esc, как ответ "Нет".

key "game\_menu" action no\_action

style confirm\_frame is gui\_frame

style confirm\_prompt is gui\_prompt

style confirm\_prompt\_text is gui\_prompt\_text

style confirm\_button is gui\_medium\_button

style confirm\_button\_text is gui\_medium\_button\_text

style confirm\_frame:

background Frame([ "gui/confirm\_frame.png", "gui/frame.png"], gui.confirm\_frame\_borders, tile=gui.frame\_tile)

padding gui.confirm\_frame\_borders.padding

xalign .5

yalign .5

style confirm\_prompt\_text:

text\_align 0.5

layout "subtitle"

style confirm\_button:

properties gui.button\_properties("confirm\_button")

style confirm\_button\_text:

properties gui.button\_text\_properties("confirm\_button")

## Экран индикатора пропуска ###################################################

##

## Экран индикатора пропуска появляется для того, чтобы показать, что идёт

## пропуск.

##

## https://www.renpy.org/doc/html/screen\_special.html#skip-indicator

screen skip\_indicator():

zorder 100

style\_prefix "skip"

frame:

hbox:

spacing 9

text \_("Пропускаю")

text "▸" at delayed\_blink(0.0, 1.0) style "skip\_triangle"

text "▸" at delayed\_blink(0.2, 1.0) style "skip\_triangle"

text "▸" at delayed\_blink(0.4, 1.0) style "skip\_triangle"

## Эта трансформация используется, чтобы мигать стрелками одна за другой.

transform delayed\_blink(delay, cycle):

alpha .5

pause delay

block:

linear .2 alpha 1.0

pause .2

linear .2 alpha 0.5

pause (cycle - .4)

repeat

style skip\_frame is empty

style skip\_text is gui\_text

style skip\_triangle is skip\_text

style skip\_frame:

ypos gui.skip\_ypos

background Frame("gui/skip.png", gui.skip\_frame\_borders, tile=gui.frame\_tile)

padding gui.skip\_frame\_borders.padding

style skip\_text:

size gui.notify\_text\_size

style skip\_triangle:

## Нам надо использовать шрифт, имеющий в себе символ U+25B8 (стрелку выше).

font "DejaVuSans.ttf"

## Экран уведомлений ###########################################################

##

## Экран уведомлений используется, чтобы показать игроку оповещение. (Например,

## когда игра автосохранилась, или был сделан скриншот)

##

## https://www.renpy.org/doc/html/screen\_special.html#notify-screen

screen notify(message):

zorder 100

style\_prefix "notify"

frame at notify\_appear:

text "[message!tq]"

timer 3.25 action Hide('notify')

transform notify\_appear:

on show:

alpha 0

linear .25 alpha 1.0

on hide:

linear .5 alpha 0.0

style notify\_frame is empty

style notify\_text is gui\_text

style notify\_frame:

ypos gui.notify\_ypos

background Frame("gui/notify.png", gui.notify\_frame\_borders, tile=gui.frame\_tile)

padding gui.notify\_frame\_borders.padding

style notify\_text:

properties gui.text\_properties("notify")

## Экран NVL ###################################################################

##

## Этот экран используется в диалогах и меню режима NVL.

##

## https://www.renpy.org/doc/html/screen\_special.html#nvl

screen nvl(dialogue, items=None):

window:

style "nvl\_window"

has vbox:

spacing gui.nvl\_spacing

## Показывает диалог или в vpgrid, или в vbox.

if gui.nvl\_height:

vpgrid:

cols 1

yinitial 1.0

use nvl\_dialogue(dialogue)

else:

use nvl\_dialogue(dialogue)

## Displays the menu, if given. The menu may be displayed incorrectly if

## config.narrator\_menu is set to True.

for i in items:

textbutton i.caption:

action i.action

style "nvl\_button"

add SideImage() xalign 0.0 yalign 1.0

screen nvl\_dialogue(dialogue):

for d in dialogue:

window:

id d.window\_id

fixed:

yfit gui.nvl\_height is None

if d.who is not None:

text d.who:

id d.who\_id

text d.what:

id d.what\_id

## Это контролирует максимальное число строк NVL, могущих показываться за раз.

define config.nvl\_list\_length = gui.nvl\_list\_length

style nvl\_window is default

style nvl\_entry is default

style nvl\_label is say\_label

style nvl\_dialogue is say\_dialogue

style nvl\_button is button

style nvl\_button\_text is button\_text

style nvl\_window:

xfill True

yfill True

background "gui/nvl.png"

padding gui.nvl\_borders.padding

style nvl\_entry:

xfill True

ysize gui.nvl\_height

style nvl\_label:

xpos gui.nvl\_name\_xpos

xanchor gui.nvl\_name\_xalign

ypos gui.nvl\_name\_ypos

yanchor 0.0

xsize gui.nvl\_name\_width

min\_width gui.nvl\_name\_width

text\_align gui.nvl\_name\_xalign

style nvl\_dialogue:

xpos gui.nvl\_text\_xpos

xanchor gui.nvl\_text\_xalign

ypos gui.nvl\_text\_ypos

xsize gui.nvl\_text\_width

min\_width gui.nvl\_text\_width

text\_align gui.nvl\_text\_xalign

layout ("subtitle" if gui.nvl\_text\_xalign else "tex")

style nvl\_thought:

xpos gui.nvl\_thought\_xpos

xanchor gui.nvl\_thought\_xalign

ypos gui.nvl\_thought\_ypos

xsize gui.nvl\_thought\_width

min\_width gui.nvl\_thought\_width

text\_align gui.nvl\_thought\_xalign

layout ("subtitle" if gui.nvl\_text\_xalign else "tex")

style nvl\_button:

properties gui.button\_properties("nvl\_button")

xpos gui.nvl\_button\_xpos

xanchor gui.nvl\_button\_xalign

style nvl\_button\_text:

properties gui.button\_text\_properties("nvl\_button")

################################################################################

## Мобильные варианты

################################################################################

style pref\_vbox:

variant "medium"

xsize 675

## Раз мышь может не использоваться, мы заменили быстрое меню версией,

## использующей меньше кнопок, но больших по размеру, чтобы их было легче

## касаться.

screen quick\_menu():

variant "touch"

zorder 100

if quick\_menu:

hbox:

style\_prefix "quick"

xalign 0.5

yalign 1.0

textbutton \_("Назад") action Rollback()

textbutton \_("Пропуск") action Skip() alternate Skip(fast=True, confirm=True)

textbutton \_("Авто") action Preference("auto-forward", "toggle")

textbutton \_("Меню") action ShowMenu()

style window:

variant "small"

background "gui/phone/textbox.png"

style radio\_button:

variant "small"

foreground "gui/phone/button/radio\_[prefix\_]foreground.png"

style check\_button:

variant "small"

foreground "gui/phone/button/check\_[prefix\_]foreground.png"

style nvl\_window:

variant "small"

background "gui/phone/nvl.png"

style main\_menu\_frame:

variant "small"

background "gui/phone/overlay/main\_menu.png"

style game\_menu\_outer\_frame:

variant "small"

background "gui/phone/overlay/game\_menu.png"

style game\_menu\_navigation\_frame:

variant "small"

xsize 510

style game\_menu\_content\_frame:

variant "small"

top\_margin 0

style pref\_vbox:

variant "small"

xsize 600

style bar:

variant "small"

ysize gui.bar\_size

left\_bar Frame("gui/phone/bar/left.png", gui.bar\_borders, tile=gui.bar\_tile)

right\_bar Frame("gui/phone/bar/right.png", gui.bar\_borders, tile=gui.bar\_tile)

style vbar:

variant "small"

xsize gui.bar\_size

top\_bar Frame("gui/phone/bar/top.png", gui.vbar\_borders, tile=gui.bar\_tile)

bottom\_bar Frame("gui/phone/bar/bottom.png", gui.vbar\_borders, tile=gui.bar\_tile)

style scrollbar:

variant "small"

ysize gui.scrollbar\_size

base\_bar Frame("gui/phone/scrollbar/horizontal\_[prefix\_]bar.png", gui.scrollbar\_borders, tile=gui.scrollbar\_tile)

thumb Frame("gui/phone/scrollbar/horizontal\_[prefix\_]thumb.png", gui.scrollbar\_borders, tile=gui.scrollbar\_tile)

style vscrollbar:

variant "small"

xsize gui.scrollbar\_size

base\_bar Frame("gui/phone/scrollbar/vertical\_[prefix\_]bar.png", gui.vscrollbar\_borders, tile=gui.scrollbar\_tile)

thumb Frame("gui/phone/scrollbar/vertical\_[prefix\_]thumb.png", gui.vscrollbar\_borders, tile=gui.scrollbar\_tile)

style slider:

variant "small"

ysize gui.slider\_size

base\_bar Frame("gui/phone/slider/horizontal\_[prefix\_]bar.png", gui.slider\_borders, tile=gui.slider\_tile)

thumb "gui/phone/slider/horizontal\_[prefix\_]thumb.png"

style vslider:

variant "small"

xsize gui.slider\_size

base\_bar Frame("gui/phone/slider/vertical\_[prefix\_]bar.png", gui.vslider\_borders, tile=gui.slider\_tile)

thumb "gui/phone/slider/vertical\_[prefix\_]thumb.png"

style slider\_vbox:

variant "small"

xsize None

style slider\_slider:

variant "small"

xsize 900