

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ «ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук

Департамент программной инженерии

**СОГЛАСОВАНО**

Доцент департамента  
программной инженерии факультета  
компьютерных наук, кандидат  
технических наук

 О. В. Максименкова

« 7 » \_\_\_\_\_ апреля \_\_\_\_\_ 2023 г.

**УТВЕРЖДАЮ**

Академический руководитель  
образовательной программы  
«Программная инженерия»,  
профессор департамента  
программной инженерии, кандидат  
технических наук

 В. В. Шилов

« 7 » \_\_\_\_\_ апреля \_\_\_\_\_ 2023 г.

**ПРОГРАММА ВИЗУАЛЬНОЙ ИНТЕРПРЕТАЦИИ ЭМОЦИЙ  
ПОЛЬЗОВАТЕЛЯ ДЛЯ ПОСТРОЕНИЯ 2D-АВАТАРА**

Текст программы

**ЛИСТ УТВЕРЖДЕНИЯ**

**RU.17701729.05.01-01 12 01-1**

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл	

Исполнитель студент группы БПИ 204

 /М. К. Аленов/

« 03 » \_\_\_\_\_ апреля \_\_\_\_\_ 2023 г.

**Москва 2023**

УТВЕРЖДЕН  
RU.17701729.05.01-01 12 01-1

**ПРОГРАММА ВИЗУАЛЬНОЙ ИНТЕРПРЕТАЦИИ ЭМОЦИЙ  
ПОЛЬЗОВАТЕЛЯ ДЛЯ ПОСТРОЕНИЯ 2D-АВАТАРА**

**Текст программы**

**RU.17701729.02.06-01 12 01-1**

**Листов 36**

Инв. № подл	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

## СОДЕРЖАНИЕ

<b>1. ВВЕДЕНИЕ.....</b>	<b>4</b>
1.1 Назначение .....	4
1.2 Область применения .....	4
1.3 Стандарты документации.....	4
<b>2. ТЕКСТ ПРОГРАММЫ.....</b>	<b>5</b>
2.1 Назначение файлов программы .....	5
ГЛОБАЛЬНЫЕ КОМПОНЕНТЫ.....	5
МОДАЛЬНЫЕ КОМПОНЕНТЫ .....	6
КОМПОНЕНТЫ, СВЯЗАННЫЕ С ЭМОЦИЯМИ.....	7
КОМПОНЕНТЫ, СВЯЗАННЫЕ СО СПРАЙТАМИ .....	7
КОМПОНЕНТЫ, СВЯЗАННЫЕ С ВИЗУАЛИЗАЦИЕЙ МАСКОТА.....	7
ИНТЕРФЕЙСЫ И ПЕРЕЧИСЛЕНИЯ.....	9
ДОПОЛНИТЕЛЬНЫЕ УТИЛИТЫ .....	9
КОМПОНЕНТЫ, СВЯЗАННЫЕ С НАСТРОЙКАМИ.....	10
2.2 Назначение методов, полей, свойств, событий файлов проекта.....	11
Файл EmotionPart.tsx: .....	11
Файл EmotionsSelection.tsx: .....	11
Файл IConf.ts: .....	12
Файл IEmotion.ts: .....	13
Файл IMascot.ts: .....	13
Файл IMascotData: .....	14
Файл IPart.ts: .....	15
Файл MascotCanvas.tsx: .....	15
Файл MascotPart.tsx: .....	16
Файл ShadowCanvas.tsx: .....	18
Файл ShadowPart.tsx: .....	19
Файл EmotionAdd.tsx: .....	19
Файл PartAdd.tsx: .....	20
Файл Proceed.tsx: .....	21
Файл ProjectAdd.tsx: .....	22
Файл ProjectAddScrap.tsx: .....	22
Файл PartPart.tsx: .....	23
Файл PartsSelection.tsx: .....	23
Файл BackgroundColorPicker.tsx: .....	24
Файл MicMinMaxDisplay.tsx: .....	25
Файл Slider.tsx: .....	26

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

Файл TwoSlider.tsx: .....	27
Файл Commands.ts: .....	28
Файл DummyMascot.ts: .....	28
Файл EDescriptor.ts: .....	29
Файл Save.ts: .....	29
Файл UUIDGen.ts: .....	30
Файл App.tsx: .....	30
Файл Projects.tsx: .....	31
3. ИСТОЧНИКИ .....	35
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ .....	36

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

## 1. ВВЕДЕНИЕ

### 1.1 Назначение

Программа предназначена для создания двумерных маскотов, являющихся визуализацией эмоций пользователя в реальном времени, используя спрайты. Программа предоставляет пользователю функционал создания, удаления и редактирования маскотов (проектов), настройку используемых эмоций для выбранного маскота, настройку используемых эмоциями спрайтов и расположения выбранных спрайтов относительно маскота.

### 1.2 Область применения

Программа разработана для создателей развлекательного видео контента, использующих или желающих использовать в своих видео аватар для репрезентации своего лица, однако не желающих обращаться к решениям, задействующим дорогие и сложные в эксплуатации трехмерные модели. Программа предоставляет пользователю возможность быстро создать аватар (маскот) под свои персональные нужды, используя двумерные спрайты.

### 1.3 Стандарты документации

Документ разработан в соответствии с ГОСТ и ЕСПД [1-8].

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

## 2. ТЕКСТ ПРОГРАММЫ

### 2.1 Назначение файлов программы

Файлы проекта можно разделить на несколько смысловых типов:

1. Глобальные компоненты.
2. Модальные компоненты.
3. Компоненты, связанные с эмоциями.
4. Компоненты, связанные со спрайтами
5. Компоненты, связанные с настройками.
6. Компоненты, связанные с визуализацией маскота.
7. Интерфейсы и перечисления.
8. Дополнительные утилиты.

### ГЛОБАЛЬНЫЕ КОМПОНЕНТЫ

Файл	Назначение
main.tsx	Точка входа в приложение, содержит инстанс компонента App.
App.tsx	<p>Основной компонент, в котором происходит взаимодействие пользователя с программой. При запуске приложения возвращает пользователю компонент Projects для выбора проектов, затем визуализирует основное окно для работы с выбранным проектом, включая в себя компоненты MascotCanvas, EmotionsSelection, PartsSelection, BackgroundColorPicker, CamSelection, MicMinMaxSelection, MicSelection, ShakeSettings.</p> <p>Также внутри компонента создается контекст с объектом IMascot, который используется внутри вышеописанных компонентов.</p>
Projects.tsx	Компонент, позволяющий пользователю выбрать/создать/удалить проект. Состоит из списка кнопок, позволяющих манипулировать созданием/выбором/удалением проектов, а также из списка загруженных проектов. Использует модальные компоненты ProjectAdd, ProjectAddScrap и Proceed. При появлении в DOM'е использует/инициализирует список проектов пользователя из файла, сохраненного в системе, и

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

	контролирует его контент.
--	---------------------------

**МОДАЛЬНЫЕ КОМПОНЕНТЫ**

Файл	Назначение
EmotionAdd.tsx	Позволяет пользователю создать/редактировать эмоцию в выбранном проекте
PartAdd.tsx	Позволяет пользователю создать/редактировать спрайт в выбранном проекте
Proceed.tsx	Представляет из себя всплывающее окно с кнопкой и текстом. Нажатие на кнопку является подтверждением того, что пользователь согласен с описанным текстом
ProjectAdd.tsx	Позволяет пользователю создать пустой проект
ProjectAddScrap.tsx	Позволяет пользователю создать проект на основе папки с подпапками, хранящими изображения, где подпапки станут эмоциями, а изображения - спрайтами

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

**КОМПОНЕНТЫ, СВЯЗАННЫЕ С ЭМОЦИЯМИ**

Файл	Назначение
EmotionPart.tsx	Является частью списка внутри EmotionsSelection. Хранит в себе информацию о эмоции в выбранном проекте. Имеет функционал изменения видимости используемой эмоции
EmotionsSelection.tsx	Визуализирует все эмоции, используемые в проекте. Содержит EmotionAdd с возможность использования в режиме редактирования или создания, также функционал выбора и удаления выбранной эмоции из проекта и список всех эмоций, реализуемых через EmotionPart.

**КОМПОНЕНТЫ, СВЯЗАННЫЕ СО СПРАЙТАМИ**

Файл	Назначение
PartPart.tsx	Является частью списка внутри PartsSelection. Хранит в себе информацию о эмоции в выбранном проекте. Имеет функционал изменения видимости используемой эмоции, а также функционал изменение положения в иерархии визуализации в окне
PartsSelection.tsx	Визуализирует все спрайты, используемые в выбранной в EmotionsSelect эмоции. Содержит PartAdd с возможность использования в режиме редактирования или создания, также функционал выбора и удаления выбранного спрайта из проекта и список всех спрайтов для выбранной эмоции, реализуемых через PartPart.

**КОМПОНЕНТЫ, СВЯЗАННЫЕ С ВИЗУАЛИЗАЦИЕЙ МАСКОТА**

Файл	Назначение	
MascotPart.tsx	Визуализирует внутри MascotCanvas изображение, хранимое в переданном при создании спрайте.  Если спрайт является активным спрайтов в PartsSelection, то данный компонент предоставляет возможность перемещать себя внутри области MascotCanvas и изменять свои размеры.	
Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

	<p>После интеракции с пользователем, компонент записывает измененное положение / размер спрайта в контекст объект IMascot из App.tsx.</p> <p>Если спрайт не является активным спрайтов в PartsSelection, то данный компонент просто возвращает изображение с указанными в спрайте (объект IPart) характеристиками (размер и позиция внутри MascotCanvas)</p>
ShadowPart.tsx	<p>Является облегченным аналогом компонента MascotPart. Различие заключается в отсутствии функционала перемещения и измене</p>
MascotCanvas.tsx	<p>При инициализации связывается с контекстом и получает из него объект IMascot. Для выбранной пользователем эмоции визуализирует на отведенной ему области компоненты MascotPart для всех спрайтов эмоции, которые пользователь оставил видимыми в компоненте PartsSelection.</p> <p>Имеет функционал изменения зума.</p> <p>Если пользователь входит в область расположения MascotCanvas, MascotPart, являющийся репрезентацией выбранного пользователем спрайта, становится доступным для редактирования.</p>
ShadowCanvas.tsx	<p>Визуализирует внутри себя объекты ShadowPart.</p> <p>При инициализации получает объект IMascot и открывает интервал, связывающийся через Tauri Api [11] с Rust частью для получения данных о состоянии маскота в формате IMascotData. Полученные данные обрабатываются внутри компонента в визуализацию маскота через изменение копии полученного объекта маскота.</p> <p>Главным отличием от MascotCanvas является единовременный рендер всех спрайтов всех эмоций, добавленных пользователем в маскота. В зависимости от получаемых с Rust'a данных неиспользуемые ShadowPart становятся невидимыми в DOM'e, что позволяет добиться высокой скорости изменения визуализации маскота в</p>

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

	графическом интерфейсе
--	------------------------

## ИНТЕРФЕЙСЫ И ПЕРЕЧИСЛЕНИЯ

Файл	Назначение
EEmotion.ts	Енумератор типов эмоций (злость, страх и т.п.)
EPart.ts	Енумератор типов спрайтов (лицо, закрытые глаза, открытые глаза и т.п.)
IConf.ts	Интерфейс, используемый для сохранения данных в конфигурационной файле проектов, используемым в Projects.tsx
IEmotion.ts	Интерфейс, используемый для описания эмоции. Является частью IMascot
IMascot.ts	Интерфейс, используемый для описания маскота
IMascotData.ts	Интерфейс, используемый для получения данных с Rust о состоянии маскота
IPart.ts	Интерфейс, используемый для описания спрайта. Является частью IEmotion

## ДОПОЛНИТЕЛЬНЫЕ УТИЛИТЫ

Файл	Назначение
Colors.ts	Описывает цвета, используемые приложением
Commands.ts	Имплементации методов, используемых для вызова Rust методов через Tauri Api [11]. Таких как: получение всех камер, выбор камеры, получение информации о состоянии маскота и т.д.

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

Config.ts	Структура данных, необходимая для корректного выбора камеры в команде из Commands.ts
DummyMascot.ts	Имплементация интерфейса IMascot с пустыми полями
EDescriptor.ts	Используется для получения иконок типов эмоций и спрайтов по передаваемым объектам EEmotion или EPart
MuiTheme.tsx	Стили для библиотеки mui/material
Save.ts	Содержит функции для сохранения и очистки директории с проектом
TextToColor.ts	Содержит функцию, возвращающую случайный цвет в формате hex по переданной строке
UUIDGen.ts	Содержит функцию генерации случайных уникальных идентификаторов

#### КОМПОНЕНТЫ, СВЯЗАННЫЕ С НАСТРОЙКАМИ

Файл	Назначение
BackgroundColorPicker.tsx	Компонент, предоставляющий функционал выбора цвета заднего фона маскота
CamSelection.tsx	Компонент, предоставляющий функционал выбора камеры. Получает список камер и устанавливает выбранную через Commands.
MicMinMaxDisplay.tsx	Компонент, предоставляющий функционал выбора минимальной и максимальной громкости микрофона
MicSelection.tsx	Компонент, предоставляющий функционал выбора микрофона. Получает список микрофонов и устанавливает выбранную через Commands.
ShakeSettings.tsx	Компонент, предоставляющий функционал изменения уровня тряски маскота в окне визуализации маскота реализуемый через Slider.tsx
Slider.tsx	Компонент, реализующий слайдер. Используется в

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

	ShakeSettings.tsx
TwoSlider.tsx	Компонент, реализующий слайдер с двумя хендлами. Используется в MicMinMaxDisplay.tsx

## 2.2 Назначение методов, полей, свойств, событий файлов проекта

Все файлы с расширением «.tsx» содержат функции, возвращающие React компонент. Описание компонентов можно найти в Приложении 1, поэтому далее при описании файлов назначение React функции не будет описано.

### Файл EmotionPart.tsx:

Название	Тип члена	Тип значения	Входные параметры	Назначение
visible	Поле	Boolean	-	Визуализирует статус видимости спрайта
setVisible	функция	React.Dispatch <React.SetStateAction <boolean>>	value: boolean	Сеттер для стейт хука
mascot	Поле	IMascot	-	Информация о загруженном проекте из контекста

### Файл EmotionsSelection.tsx:

Название	Тип члена	Тип значения	Входные параметры	Назначение
mascot	Поле	IMascot	-	Информация о загруженном проекте из контекста

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

selectIndex	поле	-	-	Отвечает за подсветку выбранной пользователем эмоции
setSelectIndex	функция	React.Dispatch <React.SetStateAction <boolean>>	value: boolean	Сеттер для стейт хука
openAdd	поле	boolean	-	Отвечает за состояние модального компонента EmotionAdd
setOpenAdd	функция	React.Dispatch <React.SetStateAction <boolean>>	value: boolean	Сеттер для стейт хука
useRedact	поле	boolean	-	Отвечает за режим модального компонента EmotionAdd (создание эмоции / редактирование)
setUseRedact	функция	React.Dispatch <React.SetStateAction <boolean>>	value: boolean	Сеттер для стейт хука
handleListItemClick	функция	-	index: number	Обрабатывает нажатие пользователя на элемент в списке эмоций

**Файл IConf.ts:**

Название	Тип члена	Тип значения	Входные параметры	Назначение
path	поле	string	-	Путь до проекта в файловой системе компьютера
name	поле	string	-	Название проекта

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

previewPath	поле	string	-	Путь до изображения, используемого как превью проекта
-------------	------	--------	---	---

**Файл IEmotion.ts:**

Название	Тип члена	Тип значения	Входные параметры	Назначение
name	поле	string	-	Название эмоции
visibility	поле	boolean	-	Видимость эмоции
parts	поле	IPart[]	-	Спрайты, привязанные к эмоции
emotion	поле	EEmotion	-	Тип эмоции

**Файл IMascot.ts:**

Название	Тип члена	Тип значения	Входные параметры	Назначение
emotions	поле	Emotion[]	-	Эмоции маскарада
bgColor	поле	string	-	Цвет фона маскарада
selectedEmotion	поле	number	-	Индекс выбранной пользователем эмоции
selectedPart	поле	number	-	Индекс выбранного пользователем спрайта

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

## RU.17701729.05.01-01 12 01-1

workingDir	поле	string	-	Путь до директории проекта
projectName	поле	string	-	Название проекта
zoom	поле	number	-	Уровень увеличения маскота в MascotCanvas и ShadowCanvas
shake	поле	number	-	Уровень тряски маскота в зависимости от громкости микрофона
maxMic	поле	number	-	Уровень ограничения по максимальной громкости микрофона
minMic	поле	number	-	Уровень ограничения по минимальной громкости микрофона

## Файл IMascotData:

Название	Тип члена	Тип значения	Входные параметры	Назначение
emotion	поле	string	-	Тип эмоции. Конвертируется в нормальный енам через EDescriptor
blink	поле	boolean	-	Состояние глаз (закрытые / открытые)
lips	поле	boolean	-	Состояние губ (закрытые / открытые)
voice	поле	number	-	Громкость микрофона

## Файл IPart.ts:

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

## RU.17701729.05.01-01 12 01-1

Название	Тип члена	Тип значения	Входные параметры	Назначение
name	поле	string	-	Имя спрайта
visibility	поле	boolean	-	Видимость спрайта
sourcePath	поле	string	-	Путь до изображения, используемого спрайтом
positionX	поле	number	-	Позиция по координате X в MascotCanvas и ShadowCanvas
positionY	поле	number	-	Позиция по координате Y в MascotCanvas и ShadowCanvas
height	поле	number	-	Высота изображения
width	поле	number	-	Ширина изображения
type	поле	EPart	-	Тип спрайта

## Файл MascotCanvas.tsx:

Название	Тип члена	Тип значения	Входные параметры	Назначение
mascot	Поле	IMascot	-	Информация о загруженном проекте из контекста
inFocus	поле	boolean	-	Навелся ли пользователь на компонент. Если пользователь навелся, то активируется функционал работы с изображениями в виде перемещения спрайта и изменения его размеров

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

setInFocus	функция	React.Dispatch <React.SetStateAction <boolean >>	value: boolean	Сеттер для стейт хука
zoomIn	функция	-		Увеличивает зум
zoomOut	функция	-		Уменьшает зум

**Файл MascotPart.tsx:**

Название	Тип члена	Тип значения	Входные параметры	Назначение
mascot	Поле	IMascot	-	Информация о загруженном проекте из контекста
currentPos	поле	{x: number, y:number}	-	Позиция компонента на координатной плоскости MascotCanvas
setCurrentPos	функция	React.Dispatch <React.SetStateAction < {x: number, y:number}>>		Сеттер для стейт хука
height	поле	number	-	Высота компонента
setHeight	функция	React.Dispatch <React.SetStateAction < number >>	value: number	Сеттер для стейт хука
width	поле	number	-	Ширина компонента
setWidth	функция	React.Dispatch <React.SetStateAction < number >>	value: number	Сеттер для стейт хука
loading	поле	boolean	-	Установлен в false, пока не будет загружено

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

				изображение в компонент
setLoading	функция	React.Dispatch <React.SetStateAction < boolean >>	value: boolean	Сеттер для стейт хука
zoom	поле	number	-	Значение увеличения из IMascot
setZoom	функция	React.Dispatch <React.SetStateAction < number >>	value: number	Сеттер для стейт хука
resizing	поле	boolean	-	True, когда происходит изменение размеров компонента. False во всех остальных случаях
setResizing	функция	React.Dispatch <React.SetStateAction < boolean >>	value: boolean	Сеттер для стейт хука
infoPos	поле	{x: number, y:number}	-	Положение информационной панели
setinfoPos	функция	React.Dispatch <React.SetStateAction < {x: number, y:number}>>	value: {x: number, y:number}	Сеттер для стейт хука
defaultHW	поле	{x: number, y:number}	-	Стартовые значения высоты и ширины
setDefaultHW	функция	React.Dispatch <React.SetStateAction < {x: number, y:number}>>	value: {x: number, y:number}	Сеттер для стейт хука
sizeHandler	функция	-	mouseDownEvent: any, dir:string	Функция изменения размеров компонента

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

handleStop	функция	-	dragElement: {x: number, y:number}>>	Функция, сохраняющая в контекст изменения позиции
------------	---------	---	--------------------------------------	---

**Файл ShadowCanvas.tsx:**

Название	Тип члена	Тип значения	Входные параметры	Назначение
mc	поле	IMascot	-	Копия маскота из контекста
setMc	функция	React.Dispatch <React.SetStateAction < >>	value: IMascot	Сеттер для стейт хука
getData	функция	-	ei: number[]	Получение данных о состоянии маскота со стороны Rust
applyDataToMascot	функция	-	Dt:IMascotData, emoIndexes: number[]	Применение данных, полученных через getData на mc внутри компонента для перерисовки

**Файл ShadowPart.tsx:**

Название	Тип члена	Тип значения	Входные параметры	Назначение
currentPos	поле	{x: number, y: number}	-	Положение компонента относительно ShadowCanvas
setCurrentPos	функция	React.Dispatch <React.SetStateAction		Сеттер для стейт хука

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

## RU.17701729.05.01-01 12 01-1

		< >>		
height	поле	number	-	Высота компонента
setHeight	функция	React.Dispatch <React.SetStateAction < number >>	value: number	Сеттер для стейт хука
width	поле	number	-	Ширина компонента
setWidth	функция	React.Dispatch <React.SetStateAction < number >>	value: number	Сеттер для стейт хука
loading	поле	boolean	-	Установлен в false, пока не будет загружено изображение в компонент
setLoading	функция	React.Dispatch <React.SetStateAction < boolean >>	value: boolean	Сеттер для стейт хука

## Файл EmotionAdd.tsx:

Название	Тип члена	Тип значения	Входные параметры	Назначение
mascot	Поле	IMascot	-	Информация о загруженном проекте из контекста
handleClose	функция	-		Закрытие компонента
handleChange	функция	-		Обработка выбора пользователем пути в файловой системе
designation	поле	EEmotion	-	Тип добавляемой эмоции
Номер изменения	Подпись ответственного за внесение изменения			Дата внесения изменения

setDesignation	функция	React.Dispatch <React.SetStateAction < >>		Сеттер для стейт хука
name	поле	string	-	Название эмоции
setName	функция	React.Dispatch <React.SetStateAction < >>		Сеттер для стейт хука

**Файл PartAdd.tsx:**

Название	Тип члена	Тип значения	Входные параметры	Назначение
mascot	Поле	IMascot	-	Информация о загруженном проекте из контекста
path	поле	string	-	Путь
setPath	функция	React.Dispatch <React.SetStateAction < string>>	value: string	Сеттер для стейт хука
designation	поле	IPart	-	
setDesignation	функция	React.Dispatch <React.SetStateAction < IPart >>	value: IPart	Сеттер для стейт хука
name	поле	string	-	Название спрайта
setName	функция	React.Dispatch <React.SetStateAction < string >>	value: string	Сеттер для стейт хука
height	поле	number	-	Высота изображения спрайта

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

## RU.17701729.05.01-01 12 01-1

setHeight	функция	React.Dispatch <React.SetStateAction < number >>	value: number	Сеттер для стейт хука
width	поле	number	-	Ширина изображения спрайта
setWidth	функция	React.Dispatch <React.SetStateAction < number >>	value: number	Сеттер для стейт хука
handleChange	функция	-	Event: SelectChangeEvent	Обработка выбора пользователем пути в файловой системе
getSizes	функция	-	Value: string	Функция, инициализирующая height и width

## Файл Proceed.tsx:

Название	Тип члена	Тип значения	Входные параметры	Назначение
handleOpen	функция	-	-	Показывает компонент
handleClose	функция	-	-	Прячет компонент

## Файл ProjectAdd.tsx:

Название	Тип члена	Тип значения	Входные параметры	Назначение
handleOpen	функция	-	-	Показывает компонент

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

## RU.17701729.05.01-01 12 01-1

handleClose	функция	-	-	Прячет компонент
name	поле	string	-	Название проекта
setName	функция	React.Dispatch <React.SetStateAction < string >>	value: string	Сеттер для стейт хука
workDir	поле	string	-	Путь до директории, в которой будет создан проект
setWorkDir	функция	React.Dispatch <React.SetStateAction < string >>	value: string	Сеттер для стейт хука

## Файл ProjectAddScrap.tsx:

Название	Тип члена	Тип значения	Входные параметры	Назначение
handleOpen	функция	-	-	Показывает компонент
handleClose	функция	-	-	Прячет компонент
destination	поле	string	-	Путь до папки, где будет инициализирован проект
setDestination	функция	React.Dispatch <React.SetStateAction < string >>	value: string	Сеттер для стейт хука
workDir	поле	string	-	Путь до папки с подпапками с изображениями
setWorkDir	функция	React.Dispatch	value: string	Сеттер для стейт хука

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

		<React.SetStateAction < string >>		
--	--	--------------------------------------	--	--

**Файл PartPart.tsx:**

Название	Тип члена	Тип значения	Входные параметры	Назначение
mascot	Поле	IMascot	-	Информация о загруженном проекте из контекста
visible	поле	boolean	-	Визуализирует статус видимости спрайта
setVisible	функция	React.Dispatch <React.SetStateAction < boolean >>	value: boolean	Сеттер для стейт хука

**Файл PartsSelection.tsx:**

Название	Тип члена	Тип значения	Входные параметры	Назначение
selectedIndex	поле	number	-	Указатель на индекс выбранного
setSelectedIndex	функция	React.Dispatch <React.SetStateAction < number >>	value: number	Сеттер для стейт хука
openAdd	поле	boolean	-	Контролирует показ компонента PartAdd
setOpenAdd	функция	React.Dispatch <React.SetStateAction	value: boolean	Сеттер для стейт хука

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

## RU.17701729.05.01-01 12 01-1

		< boolean >>		
useRedact	поле	boolean	-	Контролирует
setUseRedact	функция	React.Dispatch <React.SetStateAction < boolean >>	value: boolean	Сеттер для стейт хука
mascot	Поле	IMascot	-	Информация о загруженном проекте из контекста

## Файл BackgroundColorPicker.tsx:

Название	Тип члена	Тип значения	Входные параметры	Назначение
open	поле	boolean	-	Контроллер открытия
setOpen	функция	React.Dispatch <React.SetStateAction < boolean >>	value: boolean	Сеттер для стейт хука
handleOpen	функция	-		Функция открытия
handleClose	функция	-		Функция закрытия
color	поле	string	-	Выбранный цвет
setColor	функция	React.Dispatch <React.SetStateAction < string >>	value: string	Сеттер для стейт хука

## Файл MicMinMaxDisplay.tsx:

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

Название	Тип члена	Тип значения	Входные параметры	Назначение
min	поле	number	-	Уровень минимального звука
setMin	функция	React.Dispatch <React.SetStateAction <number>>	value: number	Сеттер для стейт хука
max	поле	number	-	Уровень максимального звука
setMax	функция	React.Dispatch <React.SetStateAction <number>>	value: number	Сеттер для стейт хука
mascot	Поле	IMascot	-	Информация о загруженном проекте из контекста
handleChange	функция	-	change:any	Контроллер изменения состояние компонента

## Файл Slider.tsx:

Название	Тип члена	Тип значения	Входные параметры	Назначение
mascot	Поле	IMascot	-	Информация о загруженном проекте из контекста
currentPos	поле	{x:number, y:number}	-	Позиция хендла слайдера
setCurrentPost	функция	React.Dispatch <React.SetStateAction < >>	value: {x:number, y:number}	Сеттер для стейт хука
drag	поле	boolean	-	True, если пользователь

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

				использует слайдер, иначе false
setDrag	функция	React.Dispatch <React.SetStateAction < boolean >>	value: boolean	Сеттер для стейт хука
normToNums	функция	number	value: number	Нормирует значение currentPos в значение от 0 до 100
numsToNorm	функция	number	value: number	Денормирует значение от 0 до 100 в currentPos
handleStop	функция	-	MouseEvent: any	Функция, срабатывающая при конце взаимодействия пользователя с хендлом слайдера. Сохраняет текущую позицию

**Файл TwoSlider.tsx:**

Название	Тип члена	Тип значения	Входные параметры	Назначение
mascot	Поле	IMascot	-	Информация о загруженном проекте из контекста
currentPosMin	поле	number	-	Позиция хендла минимальной громкости слайдера
setCurrentPostMin	функция	React.Dispatch <React.SetStateAction < number>>	value: number	Сеттер для стейт хука

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

## RU.17701729.05.01-01 12 01-1

currentPosMax	поле		-	Позиция хендла максимальной громкости слайдера
setCurrentPostMax	функция	React.Dispatch <React.SetStateAction < >>		Сеттер для стейт хука
drag	поле	boolean	-	True, если пользователь использует слайдер, иначе false
setDrag	функция	React.Dispatch <React.SetStateAction < boolean >>	value: boolean	Сеттер для стейт хука
normToNums	функция	number	value: number	Нормирует значение value в значение от 0 до 100
numsToNorm	функция	number	value: number	Денормирует значение value от 0 до 100 в currentPosMin или currentPosMax
handleStop	функция	-	MouseEvent: any	Функция, срабатывающая при конце взаимодействия пользователя с хендлом слайдера. Сохраняет текущую позицию

## Файл Commands.ts:

Название	Тип члена	Тип значения	Входные параметры	Назначение
get_mascot	функция	IMascotData	-	Invoke метода на стороне Rust'а для получения пакета информации IMascotData о состоянии маскота
get_cams	функция	String[]	-	Invoke метода на стороне Rust'а для получения списка камер, доступных

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

				пользователю
set_cams	функция	-	cam: number	Invoke метода на стороне Rust'а для установки выбранной камеры по индексу
get_mics	функция	String[]	-	Invoke метода на стороне Rust'а для получения
set_mics	функция	-	mic: number	Invoke метода на стороне Rust'а для установки выбранного микрофона по идексу

**Файл DummyMascot.ts:**

Название	Тип члена	Тип значения	Входные параметры	Назначение
DummyMascot	функция	IMascot	-	Реализация интерфейса IMascot с пустыми полями. Используется как заглушка при работе с контекстом приложения

**Файл EDescriptor.ts:**

Название	Тип члена	Тип значения	Входные параметры	Назначение
descriptPart	функция	string	part: EPart	Возвращает изображение типа спрайта
descriptEmotion	функция	string	part: emotion	Возвращает изображение типа эмоции

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

## RU.17701729.05.01-01 12 01-1

descriptRawEmotion	функция	EEmotion	emo: string	Расшифровка эмоции, полученной через get_mascot в нормальный енумератор
--------------------	---------	----------	-------------	---

## Файл Save.ts:

Название	Тип члена	Тип значения	Входные параметры	Назначение
saveMascot	функция	-	mascot: IMascot	Сохранение проекта
validateProject	функция	-	mascot: IMascot	Проверка неиспользуемых файлов проекта и удаление ненужных «.masset»
_base64ToArrayBuffer	функция	string	base64: string	Перевод строки в кодировке base64 в массив битов для сохранения изображения превью для проекта

## Файл UUIDGen.ts:

Название	Тип члена	Тип значения	Входные параметры	Назначение
generateUUID	функция	string	-	Возвращает строку, являющуюся уникальным идентификатором

## Файл App.tsx:

Название	Тип члена	Тип значения	Входные параметры	Назначение
selecting	поле	boolean	-	Если true, то пользователю показывается компонент Projects. При false - окно с

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

				выбранным проектом
setSelecting	функция	React.Dispatch <React.SetStateAction < boolean >>	value: boolean	Сеттер для стейт хука
mascot	Поле	IMascot	-	Информация о загруженном проекте из контекста
setMascot	функция	React.Dispatch <React.SetStateAction <IMascot >>		Сеттер для стейт хука
contextVisible	поле	boolean	-	Контролирует показ контекстного меню
setContextVisible	функция	React.Dispatch <React.SetStateAction < boolean >>	value: boolean	Сеттер для стейт хука
broadcasting	поле	boolean	-	True, когда происходит визуализация финальной версии маскота
setBroadcasting	функция	React.Dispatch <React.SetStateAction < boolean >>	value: boolean	Сеттер для стейт хука
value	функция	-	{ mascot, setMascot }	Объект, передаваемый в контекст. Состоит из всех компонентов хука mascot и setMascot

**Файл Projects.tsx:**

Название	Тип члена	Тип значения	Входные параметры	Назначение
----------	-----------	--------------	-------------------	------------

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

mascot	Поле	IMascot	-	Информация о загруженном проекте из контекста
dataPath	поле	string	-	Путь до основного файла конфигурации проектов приложения
setDataPath	функция	React.Dispatch <React.SetStateAction < string >>	value: string	Сеттер для стейт хука
openDialog	поле	boolean	-	Контролирует показ модального компонента ProjectAdd
setOpenDialog	функция	React.Dispatch <React.SetStateAction < boolean >>	value: boolean	Сеттер для стейт хука
openDialogAdd	поле	boolean	-	Контролирует показ модального компонента ProjectAddScrap
setOpenDialogAdd	функция	React.Dispatch <React.SetStateAction < boolean >>	value: boolean	Сеттер для стейт хука
openProceed	поле	boolean	-	Контролирует показ модального компонента Proceed при удалении проекта
setOpenProceed	функция	React.Dispatch <React.SetStateAction < boolean >>	value: boolean	Сеттер для стейт хука
projects	поле	IConf[]	-	Выгруженные из «conf.json» конфигурации

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

				проектов пользователя
setProjects	функция	React.Dispatch <React.SetStateAction < >>		Сеттер для стейт хука
selected	поле	number	-	Индекс выбранного пользователем проекта
setSelected	функция	React.Dispatch <React.SetStateAction < number >>	value: number	Сеттер для стейт хука
loadData	функция	-	-	Функция выгрузки конфигураций проектов из «conf.json»
addProject	функция	-	path: string, dir: string, name: string	Функция добавления нового проекта в «conf.json»
setProject	функция	-	proj: IConf	Чтение файла конфигурации выбранного проекта и сохранение полученного объекта IMascot в контекст приложения
searchReq	поле	string	-	Наполнение строки поиска для проектов
setSearchReq	функция	React.Dispatch <React.SetStateAction < string >>	value: string	Сеттер для стейт хука
deleteProj	поле	number	-	Индекс удаляемого проекта
setDeleteProj	Функция\	React.Dispatch <React.SetStateAction	value: number	Сеттер для стейт хука

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

## RU.17701729.05.01-01 12 01-1

		<number >>		
deleteProject	функция	-	-	Функция, удаляющая директорию с выбранным проектом
validateProject	функция	-	path: string	Функция, проверки файла конфигурации проекта
createProjectFromFiles	функция	-	path: string, destination: string	Функция, конвертирующая контент внутри папки path в проект, создаваемый в папке destination

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

Весь текст программы предоставляется в репозитории проекта <https://github.com/2Delight/mascoty-aurine>. Исходный код представлен в виде стандартных проектов/решений соответствующих сред разработки.

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

### 3. ИСТОЧНИКИ

- 1) ГОСТ 19.101-77 Виды программ и программных документов. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 2) ГОСТ 19.102-77 Стадии разработки. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 3) ГОСТ 19.103-77 Обозначения программ и программных документов. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 4) ГОСТ 19.104-78 Основные надписи. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 5) ГОСТ 19.105-78 Общие требования к программным документам. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 6) ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 7) ГОСТ 19.603-78 Общие правила внесения изменений. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 8) ГОСТ 19.604-78 Правила внесения изменений в программные документы, выполненные печатным способом – М.: ИПК Издательство стандартов, 2001.

Номер изменения	Подпись ответственного за внесение изменения	Дата внесения изменения

[illegible]