**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ**

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИ УНИВЕРСИТЕТ**

**«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук

Департамент программной инженерии

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

СОГЛАСОВАНО  
Профессор департамента программной инженерии, кандидат технических наук  
  
\_\_\_\_\_\_\_\_\_\_\_\_\_ Е.М. Гринкруг  
«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2019 г.

УТВЕРЖДАЮ  
Академический руководитель образовательной программы «Программная инженерия»  
  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.В. Шилов  
«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2019 г.

**ПРОГРАММНЫЙ КОМПОНЕНТ OPENGLVIEWER БИБЛИОТЕКИ JAVABEANS-КОМПОНЕНТ ДЛЯ 3D-ГРАФИКИ**

**Руководство оператора**

**ЛИСТ УТВЕРЖДЕНИЯ**

**RU.17701729.04.01-01 34 01-1-ЛУ**

**Исполнитель**

Студент группы БПИ163

\_\_\_\_\_\_\_\_\_\_\_\_ / Д.Е. Крайнов /

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2019 г.

**Москва 2019**

**УТВЕРЖДЕН**

**RU.17701729.04.01-01 34 01-1-ЛУ**

**ПРОГРАММНЫЙ КОМПОНЕНТ OPENGLVIEWER БИБЛИОТЕКИ JAVABEANS-КОМПОНЕНТ ДЛЯ 3D-ГРАФИКИ**

**Руководство оператора**

**RU.17701729.04.01-01 34 01-1**

**Листов 13**

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

**Москва 2019**

**АННОТАЦИЯ**

В данном документе приведено руководство оператора по эксплуатации программы «Программный компонент OpenGLViewer» (далее “OpenGLViewer”), назначением которой является работа с трехмерной графикой.

В разделе «Назначение программы» указаны сведения о назначении программы и информация, достаточная для понимания функций программы и её эксплуатации.

В разделе «Условия выполнения программы» указаны условия, необходимые для выполнения программы (минимальный состав программных и технических средств и т.п.)

В разделе «Выполнение программы» указана последовательность действий оператора, обеспечивающих загрузку, запуск, выполнение и завершение программы, приведено описание функций, формата и возможных вариантов команд, с помощью которых оператор осуществляет загрузки и управляет выполнением программы, а также ответы программы на эти команды.

В разделе «Сообщения оператору» приведены тексты сообщений, выдаваемых в ходе выполнения программы, описание их содержания и соответствующие действия оператора (действия оператора в случае сбоя, возможности повторного запуска программы и т.п.).

Настоящее руководство распространяется исключительно на программу и не заменяет учебную, справочную литературу, руководства от производителя операционной системы и прочие источники информации, освещающие работу с графическим пользовательским интерфейсом операционной системы.

СОДЕРЖАНИЕ

1. НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ 5

1.1. Функциональное назначение 5

1.2. Эксплуатационное назначение 5

1.3. Состав функций 5

2. УСЛОВИЯ ВЫПОЛНЕНИЯ ПРОГРАММЫ 6

2.1. Климатические условия эксплуатации 6

2.2. Минимальный состав технических средств 6

2.3. Минимальный состав программных средств 6

2.4. Требования к оператору 6

3. ВЫПОЛНЕНИЕ ПРОГРАММЫ 7

3.1. Запуск программы 7

3.2. Выполнение программы 7

3.3. Завершение программы 10

4. СООБЩЕНИЯ ОПЕРАТОРУ 11

СПИСОК ЛИТЕРАТУРЫ 12

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ 13

1. **НАЗНАЧЕНИЕ И ОБЛАСТЬ ПРИМЕНЕНИЯ**
   1. **Функциональное назначение**Функциональным назначением компонента является предоставление возможности графического рендеринга моделей, представленных в виде графа сцены [1], с помощью технологии OpenGL [2] и графического процессора пользовательского компьютера.
   2. **Эксплуатационное назначение**Программный компонент предлагается к эксплуатированию программистами в области компьютерной графики и разработчиками 3D-моделей в качестве инструмента для тестирования созданных графических моделей.
   3. **Состав функций**Приложение реализовано в виде набора JavaBeans-компонент [3] с возможностью автономного запуска в виде оконного приложения. Оконное приложение выполняет следующий набор функций:

* отрисовка переданных на вход компоненту OpenGLViewer графических моделей;
* загрузка модели из файла в формате json (десериализация модели);
* сохранение модели в файл в формате json (сериализация модели);
* включение/выключение отрисовки;
* включение/выключение отображения осей координат;
* включение/выключение отображения отладочного текста;
* добавление новых объектов в сцену;
* выбор текущего объекта;
* изменение объектов.

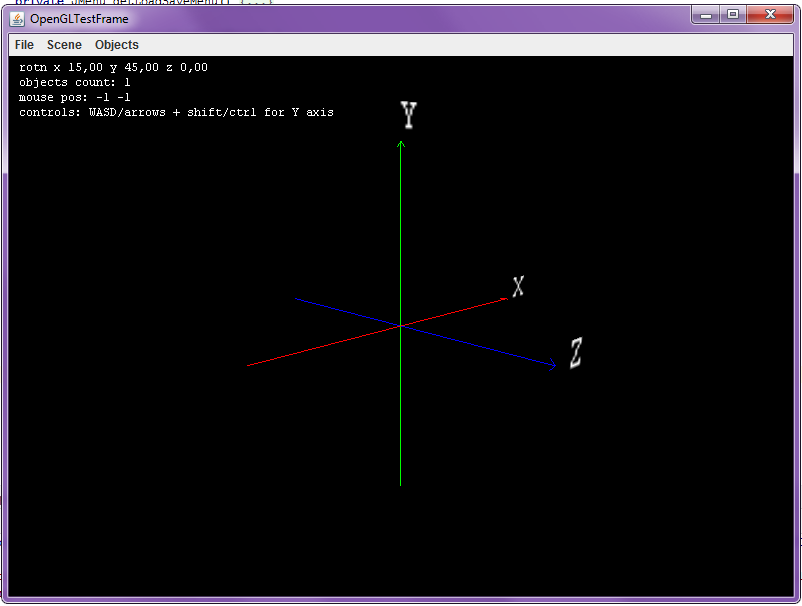
1. **УСЛОВИЯ ВЫПОЛНЕНИЯ ПРОГРАММЫ**
   1. **Климатические условия эксплуатации**

Климатические условия эксплуатации, при которых должны обеспечиваться заданные характеристики, должны удовлетворять требованиям, предъявляемым к техническим средствам в части условий их эксплуатации.

* 1. **Минимальный состав технических средств**
  + Минимальная тактовая частота процессора – 1 Ггц;
  + Минимальный объем ОЗУ – 512 Мб;
  + Минимальное свободное место на жёстком диске для хранения приложения – 10 МБ;
  + Видеокарта с минимальным размером видеопамяти 512 Мб;
  + Монитор с минимальным разрешением 800х600;
  + Клавиатура и мышь.
  1. **Минимальный состав программных средств**
  + ОС Windows XP или новее;
  + Java SE Runtime Environment 8 или новее;
  + Драйвер видеокарты с поддержкой OpenGL 2.1 или новее.
  1. **Требования к оператору**  
     Компонент эксплуатируется как составная часть библиотеки компонент. Требуемая минимальная классификация пользователя для работы в режиме просмотра – базовый оператор, для работы с библиотекой – программист.

1. **ВЫПОЛНЕНИЕ ПРОГРАММЫ**
   1. **Запуск программы**

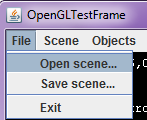
Запуск осуществляется путём открытия .jar-файла “SimpleOpenGL.jar” Должно открыться главное окно компонента. Его вид изображён на рис. 1.



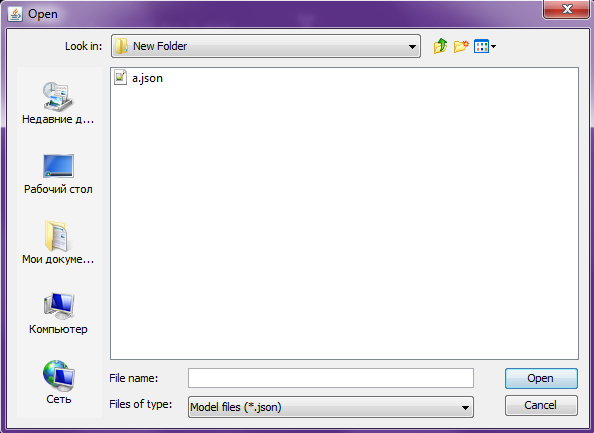
*Рисунок 1. Главное окно программы.*

* 1. **Выполнение программы**
* Загрузка модели из файла в формате json (десериализация модели)

Данное действие производится по нажатию на кнопку “Load scene…” в меню “File” программы.



*Рисунок 2. Меню “File”.*

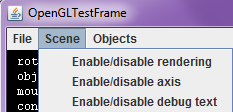
**

*Рисунок 3. Диалоговое окно выбора файла.*

Пользователь выбирает файл модели, и программа загружает его.

* Сохранение модели в файл в формате json (сериализация модели)  
  Данное действие производится по нажатию на кнопку “Save scene…” в меню “File” программы. Пользователь выбирает название файла и папку для сохранения, и программа сохраняет файл.
* Включение/выключение отрисовки

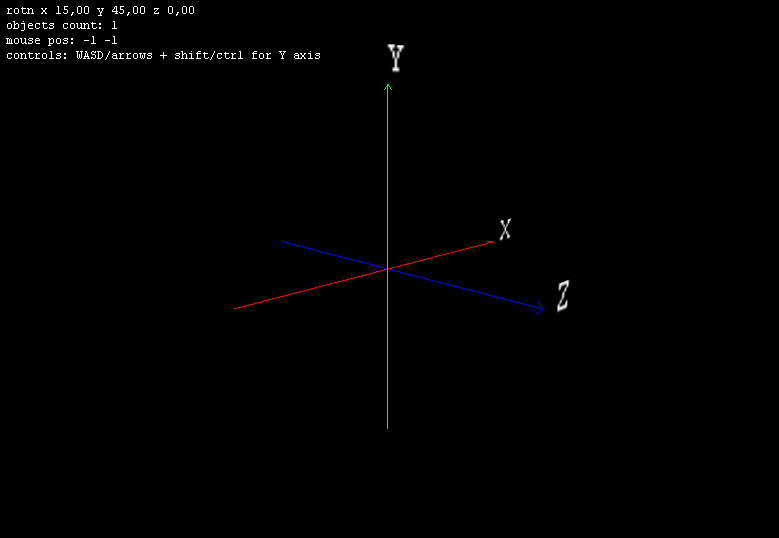
Данное действие производится по нажатию на кнопку “Enable/disable rendering” в меню “Scene” программы. По нажатию кнопки прекращается/возобновляется обновление экрана сцены.



*Рисунок 4. Меню “Scene”.*

* Включение/выключение отображения оси координат

Данное действие производится по нажатию на кнопку “Enable/disable axis” в меню “Scene” программы. По нажатию кнопки начинается/заканчивается отображение осей координат.



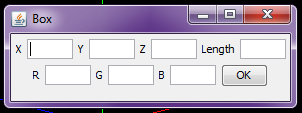
*Рисунок 5. Оси координат и отладочный текст.*

* Включение/выключение отображения отладочного текста

Данное действие производится по нажатию на кнопку “Enable/disable debug text” в меню “Scene” программы. По нажатию кнопки начинается/заканчивается отображение отладочного текста.

* Добавление новых объектов в сцену

Данное действие производится по наведению на подменю “Add child object” в меню “Objects” программы и выбору примитива из списка. Выводится окно создания объекта (рис. 6), где пользователь должен задать свойства нового объекта. По нажатию на кнопку «ОК» объект будет добавлен в граф как потомок текущего выбранного объекта.



*Рисунок 6. Пример создания объекта «Параллелепипед».*

* Выбор текущего объекта

Данное действие производится по нажатию на кнопку “Select current object…” в меню “Objects” программы и выбору примитива из списка. После выбора объекта он будет записан в свойстве окна как текущий, что позволит добавлять к нему потомков.

* Изменение объекта

Данное действие производится по нажатию на кнопку “Edit object…” в меню “Objects” программы и выбору примитива из списка. После выбора объекта будет вызвано окно, аналогичное рис. 6. Пользователь может изменить свойства объекта и сохранить либо удалить его.

* Отрисовка переданных на вход компоненту OpenGLViewer графических моделей

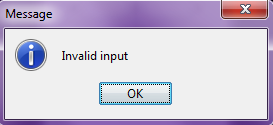
Данное действие производится автоматически, если включена отрисовка сцены (см. действие «Включение/выключение отрисовки»).

* 1. **Завершение программы**

Завершение программы производится путём нажатия кнопки “Exit” в меню “File”, либо стандартным закрытием окна в ОС Windows.

1. **СООБЩЕНИЯ ОПЕРАТОРУ**

Сообщения оператору клиентской части выводятся в виде всплывающих окон с описанием сообщения. Пример сообщения изображен на рисунке 7.



*Рисунок 10. Сообщение оператору о неверном или отсутствующем вводе свойства при создании объекта.*

Возможные сообщения:

* Invalid input (Некорректный ввод).

Выводится в случае неверного или отсутствующего числового ввода какого-либо свойства при создании или изменении объекта.

* Could not parse scene from file (Не удалось прочитать сцену из файла).

Выводится в случае невозможности прочитать модель из входного файла (см. действие «Загрузка модели из файла в формате json (десериализация модели)» в п. 3.2 данного документа.

**СПИСОК ЛИТЕРАТУРЫ**

1. Understanding and Implementing Scene Graphs [Electronic resource]. // GameDev.net [Official website]. URL: <http://archive.gamedev.net/archive/reference/programming/features/scenegraph/index.html> (accessed: 14.04.2019)

2. OpenGL 2.1 Reference Pages [Electronic resource]. // The Khronos Group Inc [Official website]. URL: <https://www.khronos.org/registry/OpenGL-Refpages/gl2.1/> (accessed: 14.04.2019)

3. JavaBeans Spec [Electronic resource]. // Oracle [Official website]. URL: <https://www.oracle.com/technetwork/articles/javaee/spec-136004.html> (accessed: 14.04.2019)

**ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Изм. | Номера листов (страниц) | | | | Всего листов (страниц) в документе | № документа | Входящий № сопроводительного документа и дата | Подпись | Дата |
| измененных | замененных | новых | аннулированных |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |