Тесхническое задание (набросок основных задач)

Основной задачей разрабатываемого программного обеспечения является предоставление удобных инструментов для разработки шейдерных программ с последующим сохранением их на языках программирования шейдеров GLSL и HLSL, которые являются одновременно наиболее популярными языками программирования шейдеров. Для выполнения поставленной задачи необходимо реализовать ряд основных модулей:

1. Модуль визуального проектирования шейдера. Данный модуль должен выполнять функции, связанный с визуальным проектированием создаваемой шейдерной программы. Под визуальным проектированием понимаются любые средства, предоставляющие возможность разработать полноценную шейдерную программу, не прибегая к написанию кода программы. Основное требование к разрабатываемому модулю – интуитивное графическое представление (понятное пользователю, не владеющему навыками программирования шейдерных программ), простота использования, наглядность (логика работы шейдерной программы должна быть понятна при просмотре визуального представления проекта).
2. Модуль текстового проектирования шейдера. Данный модуль должен выполнять функции, связанные с текстовым проектированием шейдерной программы. Сам модуль должен быть представлен в виде тактового редактора с возможностью подсветки синтаксиса шейдерного языка, авто подстановкой текста, указанием ошибок в тексте программы и прочими функциями, предлагающими более удобную разработку шейдерной программы. Для осуществления поставленной задачи должны быть реализованы текстовые статические анализаторы, которые будут анализировать исходный текст программы пользователя и, в случае появления ошибок в тексте, генерировать сообщение в виде подчёркивания и всплывающего окна (за это должна отвечать графическая часть модуля).
3. Модуль генерации исходного кода программы. Основная задача модуля – генерация текстового представления шейдерной программы, основываясь на полученных исходных данных из графического или текстового редакторов. Модуль должен выполнять свою работу как для визуализации результата, так и для сохранения результата в текстовый файл.

Кроме того, готовая программа должна реализовать следующий функционал:

1. Сохранение исходного текста шейдерной программы на любом (из доступных) языке программирования шейдеров.
2. Создание проектов для разных видов шейдерных программ:
   1. Шейдерные программы для определения внешнего вида объекта.
   2. Шейдерные программы для реализации системы частиц.
   3. Шейдерные программы для финальной обработки итоговой сцены.
   4. Прочие возможности…
3. Возможность повторного использования полученных шейдерных программ в новых проектах. Для этого необходимо реализовать собственный формат хранения данных, а также механизмы его обработки (Например, сериализация).