

ный. Вершинный шейдер – выполняется для каждой вершины в сцене (текстуры, цвет вершины, освещение вершин, факторы тумана и т. д.) Пиксельный шейдер – выполняется для каждого пикселя в цели рендеринга. Шейдеры пикселей работают совместно с шейдерами вершин; выходные данные шейдера вершин предоставляют входные данные для шейдера пикселей. Другие операции с пикселями (смещение тумана, операции с трафаретами и наложение целевого объекта визуализации) происходят после выполнения шейдера.

*DirectX*-шейдер пишется либо на языке низкого уровня, либо на *HLSL* – языке высокого уровня (похожем на *C*). Шейдеры хранятся в отдельных файлах, имеющих расширение *.hsls*. В *DirectX 11* шейдер можно скомпилировать тремя способами. Два из них используют устройство *D3D11*. Третий способ – функция *D3DCompileFromFile*, относящаяся к *HLSL*. Функция *D3DCompileFromFile* компилирует *HLSL* шейдеры. Она принимает файл с исходным кодом шейдера и возвращает экземпляр интерфейса *ID3DBlob*.

Имея скомпилированный шейдер, с помощью устройства *D3D11* необходимо создать шейдерные объекты. Для создания шейдеров разных типов интерфейс *ID3D11Device* использует разные методы, такие как *ID3D11Device::CreateVertexShader* и *ID3D11Device::CreatePixelShader*.

В ходе разработки игры были использованы вершинный и пиксельный шейдеры для создания травы. Координаты треугольников были заданы в вершинном шейдере, а затем были преобразованы для получения реалистичного эффекта. Так, для поворота каждой травинки на случайную величину по своей оси были использованы функции *rand* и *AngleAxis3x3*. При создании эффекта «движения травы по ветру» была использована встроенная переменная шейдера *\_Time* для прокрутки текстуры ветра вдоль поверхности травы.

**Ю. А. Климова**

(ГрГУ им. Я. Купалы, Гродно)

## **РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ «EQUIPMENT MANAGER» С ИСПОЛЬЗОВАНИЕМ ФРЕЙМВОРКА DJANGO**

В современном мире трудно представить рабочие места, где не используется различное техническое оборудование: планшет,

принтер, сканер и т.п. Приложение «Equipment Manager» помогает отслеживать нахождение такого оборудования у сотрудников одной компании.

Для учёта оборудования необходима информация о самом оборудовании, его владельцах, производителе, виде, а также статусе. Помимо этого, нужна информация том, когда и кто им владел. Данное приложение представляет собой взаимодействие с базой данных, где собрана вышперечисленная информация. Данное приложение представляет собой взаимодействие с реляционной базой данных, где собрана вышперечисленная информация. Для этого было создано 6 таблиц: Equipment, Human, Type, Producer, Status и Ownership. Таблицы Equipment и Human имеют между собой связь многие ко многим. Эту связь обеспечивает таблица Ownership. Также приложение обладает следующим функционалом: просмотр данных; добавление данных с помощью форм; редактирование данных; удаление данных; поиск данных; получение статистики о востребованности того или иного оборудования.

При разработке веб-приложение были использованы язык программирования Python, фреймворк Django и СУБД PostgreSQL. Данные технологии были выбраны по причине простоты использования и огромного количества инструментов для создания веб-приложений (ORM, панель администратора, пользователи, миграции). Дизайн веб-страниц был создан с помощью HTML и CSS.

Приложение «Equipment Manager» могут использовать сотрудники IT-компаний, университета, государственных учреждений и других предприятий, где широко используется техническое оборудование.

**Р. В. Ковалёв, М. В. Москалева**  
(ГГУ им. Ф. Скорины, Гомель)

## **РАЗРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ ОБУЧЕНИЯ И КОНТРОЛЯ ЗНАНИЙ ПО ДИФФЕРЕНЦИАЛЬНЫМ УРАВНЕНИЯМ**

На текущем уровне развития технологий программирования происходит внедрение их во все сферы человеческой деятельности. Современное общество очень нуждается в высококвалифицированных специалистах, у которых есть умения и навыки эффективно и грамотно использовать новые информационные технологии в разных