

2. Определены алгоритмические и программные средства для целей исследования.
 3. Разработана структура информационной системы с технологией интерактивной визуализации средствами дополненной реальности.
 4. Разработаны проекты для некоторых школьных тем, где приемлемо использование трёхмерных моделей для лучшего усвоения учебного материала и повышения мотивации при изучении темы.
- Результаты данной работы подтверждают большое значение дополненной реальности в развитии разумного и обоснованного взаимодействия человека и компьютера.

Список цитируемых источников:

1. Дополненная реальность на «живых 3D-метках» [Электронный ресурс]. — Режим доступа : <https://dic.academic.ru/dic.nsf/guwiki/1494589>. — Дата доступа : 06.10.2022.
2. Кравцов, А. А. Исследование и разработка информационной системы с технологией интерактивной визуализации средствами дополненной реальности : дисс. на соискан. уч. ст. к. техн. н. / А. А. Кравцов. — Краснодар : КубГАУ, 2016. — 167 с.
3. Интерактивные технологии EligoVision [Электронный ресурс]. — Режим доступа: <http://eligovision.ru/ru/about/what/>. — Дата доступа : 06.10.2022.
4. Караказова, А. П. Инновационные коммуникации в государственных и коммерческих структурах: технологии дополненной и виртуальной реальности / А. П. Караказова. — СПб. : СПбГУ, 2018. — 114 с.

УДК 004.42

А. Н. Коваль, Е. Г. Шапович

Учреждение образования «Барановичский государственный университет», Барановичи, Республика Беларусь

РАЗРАБОТКА ИНТЕЛЛЕКТУАЛЬНОЙ ИГРЫ «ВИКТОРИНА» СРЕДСТВАМИ C++BUILDER

Введение. Языки программирования для создания игр в настоящее время являются востребованными и развитыми в современном мире. Большинство игр на мобильные или компьютерные устройства написаны на различных языках программирования. Большая популярность игр на компьютеры, Android и iOS устройства и другие системы делают языки программирования для создания игр ценным товаром на рынке.

Игровая индустрия начала сильно развиваться в начале 2000-х годах. Когда пользователям стали доступны мощные по тому времени и относительно недорогие компьютеры. Сейчас игры на компьютерах и смартфонах остаются популярными у детей и взрослых.

В настоящее время существует множество языков программирования и большинство из них подходят для написания игр на различные платформы. Наибольшее распространение для написания игр получили следующие языки: C#, C++, Java, JavaScript,

Выше описанные языки программирования подходят для написания игр на различные платформы, так как они способны дать программисту максимальный набор инструментов и хорошо взаимодействуют с программами (Unity 3D, Game Maker, Unreal Development Kit и другие), которые нужны для разработки игр. Можно использовать и другие языки программирования, но они работают только на отдельных участках кода при написании игры [1].

Основная часть. Целью данного исследования является разработка интеллектуальной игры «Викторина» на языке C++.

Данная игра должна позволять компьютеру задавать вопросы, а человеку отгадывать давать ответы на поставленные вопросы. Предварительно программа объясняет правила игры и позволяет уточнить их в любой момент. Тематикой вопросов могут быть по выбору города, животные, растения и т. п. Тематику из предложенных компьютером (не менее 3) выбирает человек. Для игры компьютер использует собственную базу данных (для каждой тематики свою), хранящуюся в виде текстового файла. Если названный человеком ответ на вопрос неправильный (отсутствует в базе), то предлагается еще одна попытка. В конце игры подводится итог.

Для решения поставленной задачи был выбран программный продукт C++ Builder.

C++ Builder — программный продукт, инструмент быстрой разработки приложений, интегрированная среда программирования, система, используемая программистами для разработки программного обеспечения на языках программирования Си и C++ [2].

Сначала формируем алгоритм работы игры. Пользователю необходимо пройти обучение (открывается форма с объяснением основных моментов игры), обеспечить возможность узнать рейтинг игроков (на этой форме выводятся имена игроков и их очки в убывающем порядке) и смены игрока (вызывается форма, на которой можно добавить, выбрать и удалить игрока). Данные действия можно будет вызвать с помощью кнопок. После запуска игры, можно выбрать тематику викторины. После выбора тематики открывается форма с викториной и выводятся вопросы по выбранной тематике, которые берутся из файлов. В этой форме находится функция, которая проверяет введенный текст с ответом, который находится в файлах.

В основной форме, где будет проходить викторина, будут выводиться вопросы с соответствующей картинкой. А чтобы вопросы каждый раз не повторялись, предварительно будут сгенерированы случайные числа без повторений. Вопросы будут выводиться в соответствии с сгенерированной последовательностью.

Также необходимо реализовать функцию сохранения результата прохождения викторины. Блок-схема функции, которая подсчитывает количество очков показана на рисунке 1. Результат сохраняется в переменную `rezult`. Переменная `popitka` отвечает за количество попыток, который использовал игрок во время прохождения викторины. Переменная `rig` отвечает за количество правильных ответов. Переменная `t` отвечает за время, которое потратил игрок, проходя викторину. Результат сохраняется в текстовый документ.

После реализации всех заявленных требований была разработана компьютерная игра. После запуска исполняемого файла откроется главное меню, представленное на рисунке 2.

При первом запуске игры нужно добавить игрока. Нажав на кнопку «Сменить игрока». Откроется окно, представленное на рисунке 3.

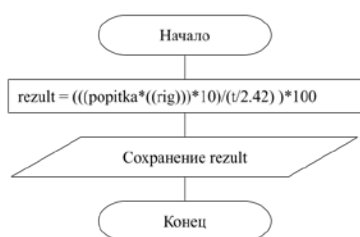


Рисунок 1 — Блок-схема алгоритма подсчёта количества очков в конце викторины



Рисунок 2 — Главное меню



Рисунок 3 — Окно «Выбор игрока»

После выбора игрока можно начать игру. Теперь мы можем начать проходить викторину. Для этого необходимо нажать на кнопку «Начать игру». Открывается окно, в котором необходимо выбрать тематику (рисунок 4).

После выбора тематики будет сгенерирована случайная последовательность вопросов из файла, после чего первый вопрос выведется на экран (рисунок 5).

Чтобы ответить на вопрос, необходимо ввести ответ в строку, и нажать на кнопку «Ответить». При правильном ответе выводится соответствующее мотивация. Вопрос засчитывается как правильный. При неправильном ответе, открывается окно, в котором предлагается перейти на следующий вопрос (ответ не будет засчитан как правильный) или можно использовать дополнительную попытку. В начале каждой игры, даётся 5 попыток.

После того как закончатся все вопросы, выводится ваш результат (рисунок 6), который можно сохранить. Результат сохраняется в файл.

Чтобы посмотреть рейтинг всех игроков, нужно нажать на кнопку «Рейтинг» на главной форме.



Рисунок 4 — Выбор тематики

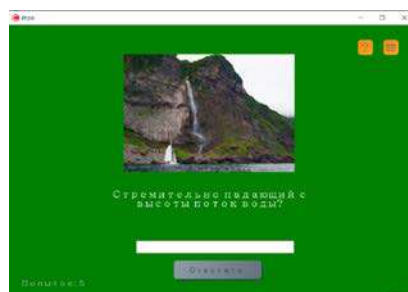


Рисунок 5 — Вывод вопроса



Рисунок 6 — Результат игры

Заключение. В ходе исследования была разработана компьютерная игра «Викторина». Данная игра предусматривает возможность создания нескольких игроков, и сохранения прохождения викторины в файлах. В игре предусмотрена возможность выбора различных тематик викторины, сохранения результатов викторины и просмотра рейтинга игроков. Также разработанная викторина позволяет сделать несколько попыток на вопрос, в случае, если ответ на вопрос неверный. Все поставленные цели в исследовании были выполнены. Также были изучены возможности среды программирования C++ Builder и объектно-ориентированного языка программирования C++.

Список цитируемых источников

1. *Моррисон, М.* Создание игр для мобильных телефонов / М. Моррисон., пер. с англ. — М. : ДМК Пресс, 2008. — 503 с.
2. C++ Builder : Software Overview Embarcadero [Электронный ресурс]. — Режим доступа : <https://www.embarcadero.com/products/cbuilder>. — Дата доступа : 06.10.2022.

УДК 004.942

М. А. Кононович, А. В. Кульша, П. П. Люцко

Учреждение образования «Барановичский государственный университет», Барановичи, Республика Беларусь

РЕДАКТИРОВАНИЕ ПОЛИГОНАЛЬНОЙ МОДЕЛИ КРЫШКИ РЕМНЯ ГРМ В СРЕДЕ BLENDER

Введение. 3D-сканирование представляет собой процесс создания виртуальной поверхностной копии детали. 3D-сканер — это очень полезное устройство, которое позволяет превратить объемную фигуру в ее электронную модель. Такую модель можно загрузить в трехмерный редактор, модифицировать или усовершенствовать. Технология 3D-сканирования одна — отбрасывается маска (шаблон) на поверхность объекта.

По искривлению маски на поверхности программное обеспечение (ПО) определяет его геометрию.

3D-сканирование открывает множество возможностей. Просто представьте, для всего того, что можно найти в реальном мире, можно получить цифровую модель всего за несколько минут.

Каждый день, тысячи компаний используют 3D-сканирование и программное обеспечение для [1]:

- создания, обратного проектирования CAD моделей реальных изделий с утерянными чертежами, а также для моделирования нового дизайна;
- контроля качества изделий путем сравнения их геометрии с эталонной CAD моделью;
- производства кастомизированных деталей и продуктов для медицины, стоматологии и индустрии рекламы и моды;
- сканирования зданий, промышленных помещений для создания актуальных моделей строений.

Основная часть. Целью данного исследования является восстановление полигональной 3D модели, полученной в результате сканирования крышки ремня ГРМ.

Актуальность задачи заключается в отказе от создания новой твердотельной модели объекта и получения модели максимально приближенной к реальной, методами восстановления и доработки полигональных моделей в программной среде Blender.

Предметом исследования является 3D-сканированная модель крышки ремня ГРМ и способы восстановления модели полученной при сканировании объекта.

При выполнении данного исследования были поставлены следующие задачи:

1. Отсканировать и получить полигональную модель крышки ремня ГРМ.
2. Изучить методы редактирования полигональной модели, полученной путем сканирования реального объекта.
3. Привести к нормальному виду полигональную модель.

Для восстановления полигональной модели могут быть использованы такие программные обеспечения:

1. 3dsMax;
2. Cinema 4D;
3. Blender.

В качестве ПО был выбран Blender, так как он используется для создания трёхмерной компьютерной графики, включающий в себя средства моделирования, рендеринга, постобработки и т. д.

На рисунке 1 представлена модель, полученная после сканирования.



Рисунок 1 — Отсканированная модель