

операций покупки товара, выполнение регистрации пользователя, отображение истории покупки товара, занесение товара в корзину пользователя, редактирование профиля, предоставление административных привилегий.

Интерфейс приложения имеет адаптивный дизайн, что позволяет пользователю комфортно использовать функции на мобильном и десктопном устройстве через браузер.

В качестве хранилища данных о пользователях, играх, предметах и услугах используется база данных, созданная и управляемая средствами СУБД phpMyAdmin. Данная СУБД считается подходящим решением для сложных операций с большими объемами данных.

В. А. Талатай, Е. В. Комракова
(ГГТУ им. П. О. Сухого, Гомель)

ИСПОЛЬЗОВАНИЕ КОМПОНЕНТА RIGIDBODY ПРИ СОЗДАНИИ ИГРЫ В UNITY

Любой игровой объект должен содержать в себе твёрдое тело, чтобы быть подверженным гравитации, действовать согласно назначенным путём скриптинга силам, или взаимодействовать с другими объектами через физический движок *NVIDIA PhysX*. Компонент *rigidbody* определяет игровой объект как физическое тело и используется для реалистичного перемещения твёрдых тел, на которые воздействуют сила вращения и другие силы.

Когда объект находится под управлением физики, он перемещается частично независимо от своих родителей. Если переместить одного из родителей, они потянут за собой *Rigidbody* потомков. Однако, твёрдые тела также будут падать вниз под воздействием силы тяжести и реагировать на события столкновений.

Чтобы контролировать твёрдые тела, в основном можно использовать скрипты для добавления сил и вращения. Также можно это сделать, вызвав на твёрдом теле (*Rigidbody*) объекта *AddForce()* и *AddTorque()*, что при использовании физики не должна напрямую меняться трансформация объекта.

Игровой проект представляет собой бегущего человечка, который должен уклоняться от катящихся шаров, и за каждый мимо пролетевший шар будут начисляться очки на счёт игрока.

В проекте была использована физика, чтобы придать динамику и коллизию игровым объектам. Галочка *Use Gravity* поставлена с целью имитации силы притяжения. *Constraints* никакие не ставятся, так как все действия игры происходят в 3D пространстве. Поле массы выставлены для объекта *Human 75*, а для катящихся шаров – 500.

Компонент *Rigidbody* очень помогает в проекте, потому что он содержит большую часть логики физических вычислений и много полезных методов, которые помогают рассчитывать коллизии, силу трения с поверхностью, задавать тип силы (импульсная, игнорирующая массу, мгновенная, постоянная). В проекте чаще всего использовался метод *AddForce()* для изменения действующей силы на игровые объекты, при этом тип силы, который используется для изменения силы, есть импульсная. Это задаёт некоторый необычный поведение, с рывками, что делает игру довольно весёлой и не такой простой.

Р. В. Топольцев
(ГГУ им. Ф. Скорины, Гомель)

ИГРОВОЕ ПРИЛОЖЕНИЕ «ТРИ В РЯД» В СРЕДЕ UNITY 3D

Игры в жанре «Три в ряд» не являются чем-то новым и уже давно, достаточно прочно, закрепились в игровой индустрии. Они характеризуются тем, что их игровое поле состоит из сетки элементов, а задачей игрока является манипулирование элементами таким образом, чтобы создавать заданные игрой шаблонные комбинации, и тем самым заставить собранные элементы исчезнуть.

Большой особенностью игр «Три в ряд», которая сильно влияет на их успех – такие игры абсолютно не требовательны ко времени. В них можно поиграть 5 минут, например, во время перерыва, или во время поездки в транспорте, а можно и вовсе увлечься, забыв про счёт времени.

Для разработки игры был выбран язык программирования C#, а игровым движком стала межплатформенная среда разработки компьютерных игр Unity, которая имеет чрезвычайно производительный визуальный рабочий процесс и мощную межплатформенную поддержку. Благодаря этому разработанное приложение можно без проблем адаптировать как для мобильных устройств, так и для персональных компьютеров.