

ХI Межрегиональная научно-практическая конференция
«НОВОЕ ПОКОЛЕНИЕ»

Секция: Архитектура и дизайн

**Разработка дизайна компьютерной игры
средствами визуального программирования**

Сидельникова Марина,
5 класс (11 лет),
МБУДО ДТДМ
г.о. Тольятти

Руководитель:

Сидельникова Мария
Александровна, педагог
дополнительного образования
МБУДО ДТДМ г.о.
Тольятти

Самара 2021

Содержание

Введение.....
Глава 1. Компьютерные игры и геймдизайн	
1.1. Развитие компьютерных игр и геймдизайн.....
1.2. Виды геймдизайна.....
1.3. Этапы создания компьютерных игр.....
Глава 2. Опытнo-экспериментальная работа по созданию компьютерной игры	
2.1. Результаты анкетирования.....
2.2. Критерии выбора и обзор конструкторов для создания игр.....
2.3. Интерфейс и основы работы в среде визуального программирования.....
2.4. Разработка дизайн компьютерной игры.....
Заключение.....
Список литературы.....

Введение

Актуальность. Компьютерные игры с каждым годом становятся все популярнее. Компьютерные игры разрабатываются для различных устройств: компьютеры, смартфоны, планшеты, консоли. Согласно исследованию аналитической компании DFC Intelligence, к середине 2020 года количество людей, так или иначе увлекающихся видеоиграми, превысило три миллиарда. Это больше трети населения всей планеты — в июле 2020 года население Земли оценивали в 7.8 миллиардов человек. Рынок игрового программного обеспечения показал огромный рост за последние годы теперь гораздо больше людей занято в развитии этого рынка. Игровая индустрия сейчас по объемам продаж соревнуется с другими формами развлечений [5, с.45].

Самым быстрорастущим сегментом игроков является мобильный — на смартфонах играет более 40% всех игроков. 48% игроков играют на персональных компьютерах, и только 8% всех игроков (около 250 миллионов) играют на консолях и регулярно покупают для них новые игры. Статистика игроков в России за 2018 год довольно внушительная. Компьютерными играми интересуется свыше 43 млн. человек. Из них 61% играет на персональных компьютерах или ноутбуках. Смартфоны и планшеты используют 57% игроков, а игровые консоли - 12%.

Наиболее активными пользователями компьютерных игр являются дети. Многие из них имеют различные устройства, на которых установлены игры. Но родители не всегда положительно относятся к увлечению компьютерными играми.

Проблема: Дети увлечены различными компьютерными играми, и не задумываются над тем, что могут применить свои знания и навыки для разработки дизайна собственных компьютерных игр.

Гипотеза: ребенок может применять свои знания, навыки и накопленный опыт в области компьютерных игр, для создания своих собственных IT-проектов - компьютерных игр, без использования сложных языков программирования

Цель: самостоятельно разработать компьютерную игру без использования сложных языков программирования

Задачи:

1. узнать историю компьютерных игр;
2. рассмотреть виды геймдизайна;
3. рассмотреть этапы создания игры;
4. провести анкетирование детей в возрасте 10-12 лет и проанализировать полученные данные;
5. разработать критерии выбора конструктора для создания игр;
6. проанализировать, конструкторы для создания игр и их возможности для разработки дизайна игры;
7. выбрать платформу для создания игры, отвечающую разработанным нами критериям;
8. разработать дизайн игры и запрограммировать объекты;
9. проанализировать результаты исследований и сделать выводы.

Глава 1. Компьютерные игры и геймдизайн

1.1. Развитие компьютерных игр и геймдизайн

Компьютерная игра – компьютерная программа, служащая для организации игрового процесса (геймплея), связи с партнёрами по игре, или сама выступающая в качестве партнёра. Термин «Компьютерная игра» означает разновидность мультимедийного продукта, представляющего собой комплексное, синтетическое произведение, созданное на особом уровне сотрудничества коллектива авторов [1, с. 137].

В настоящее время, в ряде случаев под термином «компьютерная игра» может использоваться термин «видеоигра», то есть данные термины могут употребляться как синонимы и быть взаимозаменяемыми [1, с.135]. В компьютерных играх, как правило, игровая ситуация воспроизводится на экране дисплея или обычного телевизора (в этом случае компьютерные игры одновременно являются и видеоиграми).

Компьютерные игры возникли в XX веке, их история длится более 50 лет. Первые игры появились не как развлечение, а для научных целей, поскольку первые компьютеры были огромными, дорогостоящими и в основном появлялись в образовательных и научных учреждениях. Компьютерные игры начали распространяться среди обычных пользователей с появлением консолей и первых персональных компьютеров, когда они стали более доступными для большинства людей. Только после популяризации компьютерных игр они стали коммерческими. В настоящее время данный сектор является наиболее привлекательным для компаний в коммерческом плане.

Первое подобие компьютерной игры было разработано в 1942 году. Это был ракетный симулятор. С этого времени началась эра компьютерных и видеоигр.

Основной этап развития первых компьютерных игр приходится на 50-60-е годы XX века. В данный период создавались игры, которые были предназначены не для развлечения, а как научное исследование. Игра ОХО (аналог крестов-ноликов) была удостоена докторской диссертации А. С. Дугласа. Из-за того что геймплей первых игр типа воспроизводился на черно-белых экранах осциллографов, визуальная их составляющая была предельно примитивна или полностью отсутствовала [4, с.139]. Но, несмотря на это, игры стали очень популярными среди студентов и посетителей учебных заведений. Например, такая игра - «Теннис для двоих» (аналог пинг-понга для двух игроков), созданная в 1958 году Уильямом Хигинботамом. Игры тех времен были разработаны специально для каждого компьютера и выглядели не так как современные. В качестве экрана использовались текстовые терминалы и векторные графические дисплеи.

Следующий период охватывает с 1961 по 1970-е гг. В это время была создана игра SpaceWar для компьютеров и первый прототип компьютерной мыши. В этот период развития компьютерных игр не существовало такого понятия, как геймдизайн. На данном этапе разработкой игр занимались лишь

программисты, но это продолжалось до того, как индустрия видеоигр начала стремительно расширяться в 1970-х.

Особую роль в развитии гейм-дизайна в частности и игровой индустрии в целом сыграли аркадные игры, первая из которых «Computer Space» была создана Т. Дабни и Н. Бушнеллом в 1971 году.

1971-1980 гг. – в этом десятилетии человечество сделало огромный шаг в развитии компьютеров, и люди поняли, что развитие игры - огромный источник дохода. В это время появляются персональные компьютеры и игровые консоли, появляется прототип локальной сети и Интернет. Игроки теперь могут играть в компании, объединяясь в группы, используя сеть. Этот период характеризуется тем, что задачи всей команды разработчиков довольно часто решал один человек, который совмещал в себе художественные навыки, навыки программирования и написания сценариев.

Главные события:

1971 г. – появилась первая коммерческая игра Galaxy.

1973 г. – первый в мире шутер от первого лица Mazewar появляется, с возможностью играть через сеть.

1977 г. – с появлением Apple II выпущены первые графические игры.

1979 г. – Создана первая многопользовательская игра

1980 г. – появляется игра Pac-Man, которая позже попала в Книгу рекордов Гиннеса как самая популярная игра в мире.

В период с 1981 по 1990 год консоли испытывают кризис, потому что персональные компьютеры становятся доступными из-за низкой цены.

1984 г. – появился первый космический симулятор с элементами торговли – игра Elite. Также выпущена игра King's Quest – первый в мире анимационный квест.

К середине 80-х годов распространенность аркадных видеоигр, в рамках которых выработались новые направления и жанры, стала практически повсеместной. Аркадные автоматы при этом стали более производительными и визуально более привлекательными [4, с. 139].

1985 г. – выпущена знаменитая игра «Тетрис», созданная Алексеем Пажитновым. Компания Commodore создает персональный компьютер Amiga. В этом году мир встречает братьев Марио (Super Mario Bros).

1988 г. – Конференция разработчиков игр – первая в мире конференция разработчиков игр, прошедшая в Сан-Хосе.

1989 г. – Появились процессоры Intel 486, благодаря которым был произведен переход персональных компьютеров на графический интерфейс. Выпущены такие известные игры, как SimCity, Prince of Persia, Solitaire Klondike, которые стали доступны во всех версиях Windows. Появляется первая карманная консоль.

Только к концу 80-х годов XX века многие игровые проекты приобрели такие масштабы, что потребовалось увеличивать число исполнителей и разделять полномочия в авторском коллективе. Кроме того, появился спрос на более сложные игры, что и привело к появлению такого направления как геймдизайн.

1991 - 2003 гг. – в этом десятилетии произошла революция в компьютерных играх, благодаря развитию компьютерных технологий. Стало возможным создавать миры с 3D-графикой, используя сложные мультимедийные инструменты. Игроки теперь могут сами изменять окружающий игровой мир и самого героя. Игровая индустрия начинает приносить высокие доходы, которые становятся сравнимыми с доходами киноиндустрии.

1991 г. – выпущены известные игры: Lemmings и Sonic the Hedgehog. Персонаж игры «Синий еж» позже становится символом Сеги.

1992 г. – впервые появился кровавый боевик Мортал Комбат, создавший целую серию не только игр, но и фильмов и мультфильмов.

1993 г. – Id Software выпускает культовую игру Doom – первый в мире 3D-шутер.

1995 г. – выпуск первых частей знаменитых игр Need for Speed и Heroes of Might and Magic.

1999 г. – Intel выпускает процессор Pentium III с возможностью обработки 3D-графики и потоковой передачи мультимедиа.

2000 г. – элементы искусственного интеллекта появляются в игровых движках.

2003 г. – состоялся первый чемпионат мира по электронному спорту с участием около 150 тысяч игроков со всего мира.

В 2000 е этот период игры становились более сложными, а компьютеры и игровые консоли стали мощнее, что способствовало отделению работы гейм-дизайнера от ведущего программиста. Вскоре дизайн игр стал одним из главных процессов в разработке игр. В это время и стали создаваться группы специалистов, сосредоточенные на дизайне игры.

2004 г. – следующий этап шаг в развитии компьютерных и видеоигр – развитие современных игр. В последующие годы и до сегодняшнего дня существует огромное количество выходов компьютерных и видеоигр различных жанров и для любой платформы. Развитие компьютерной техники также не стоит на месте. Важный аспект, связанный с видеоиграми – их социальная коммуникативность, возможность играть в группе, обмениваться информацией о видеоиграх в сетях, форумах, чатах, используя при этом мобильную связь [9, с. 143].

Теперь в крупных компаниях, каждый аспект игры (контроль, дизайн уровней) может иметь отдельного продюсера, ведущего дизайнера и нескольких генеральных конструкторов.

1.2. Виды геймдизайна

Игровой дизайн (также геймдизайн, англ. game design) – процесс создания формы и содержания игрового процесса (геймплея) разрабатываемой игры. Работа с геймдизайном может происходить как через соответствующий документ (англ. design document), так и существовать только в сознании разработчиков игры.

Игровой дизайн определяет: набор возможных вариантов, из которых игрок может выбирать во время игры; условия победы и поражения; как игрок контролирует происходящее в игре; как взаимодействует с игровым миром; сложность игры и др.

В геймдизайне существуют специализации:

Системный дизайн – создание правил и сопутствующих расчетов для игры – это единственная задача из области геймдизайна, актуальная для любой игры, потому что правила есть у всех игр.

Контент-дизайн – создание персонажей, предметов, загадок и миссий. Хотя он и более распространен в видеоиграх, ролевые и коллекционные карточные игры также задействуют значительное количество контента.

Дизайн уровней – создание уровней игры, включающей ландшафт карты и расположение на этой карте объектов. Хотя дизайн уровней и является широко распространенным – мастера в настольных ролевых играх составляют карты подземелий, начиная с 1970-х годов – говоря «дизайнер уровней», чаще всего имеют в виду дизайнера уровней для видеоигры.

Дизайн мира (в ММО и открытых мирах) с продумывание пространств, локаций, как они диктуют пользовательский опыт и увязываются с общей задумкой игры.

1.3. Этапы создания компьютерной игры

Основные этапы разработки видеоигры можно представить в виде основной базовой схемы:

Идея.

Исследование.

Планировка карты (уровней).

Игровые элементы.

Геометрия и моделирование.

Текстурирование.

Звуковые эффекты и сопровождение.

Бета релиз и тестирование [2, с. 3].

Рассмотрим данные пункты подробнее. Проектирование игры начинается с определения цели. Необходимо решить, к какому результату должен прийти игрок в итоге. Нельзя начать создание игры без основной идеи, которая определяет всю игру с самого начала. Чем оригинальнее будет задумка, тем больше шансов, что игра понравится пользователям.

Определение жанра игры также имеет большое значение в разработке проекта. Жанр можно корректировать по ходу создания игры, но изменить полностью его невозможно. Жанр должен быть один от начала игры и до конца.

Следующим этапом является сеттинг. Сеттинг – это художественное оформление игры, которое непосредственно влияет на ее правила.

К средствам можно отнести программный код, т. е. выбор языка программирования и дальнейшая работа с ним, а также к средствам относится игровой движок.

Творческая составляющая разработки игры имеет определяющую роль в ее создании. Игровая механика является важным компонентом системы создания игр. Строгого определения этому термину нет. По сути игровая механика – это некий свод правил, по которым работает игра. Также можно сказать, что игровая механика – правила, которые реализуют интерактивное взаимодействие игрока и игры. Игровые механики могут быть:

– простыми — элементарное действие, которое совершает игрок в игре, к примеру — бросок кубика или перемещение в пространстве из одной точки в другую.

– составными — состоят их нескольких простых. Основой игровой механики являются объекты. Игровой механикой управляет пользователь, а существует еще физический движок, отвечающий за те действия, которые происходят без прямого участия пользователя.

Разработка уровней. Уровень представляет собой определенную локацию, к примеру, город, лес, помещение и т. п. Уровни могут использоваться для того, чтобы увеличить продолжительность игры, позволяя победившему игроку играть снова, но на повышенной сложности, например, с большим количеством препятствий или с увеличенной скоростью. С таким подходом игра дольше остается интересной. Однако разделение на уровни не обязательно, есть разработчики, которые наоборот стремятся делать игру без явного перехода между уровнями, создавая впечатление непрерывности.

Оформление. Изначально создают образы всех объектов (главные и второстепенные герои, фоны и т. п.), после на их основе создают 2D или 3D-модели, далее для объектов, которые должны передвигаться в ходе игры создается анимация, а для других объектов, например, для бонусов, создаются спецэффекты. Отдельно отметим меню. Начальное меню считается визитной карточкой игры, поэтому оно тоже должно быть красиво оформлено.

Одно из главных в создании игры – это ее сюжет. Пользователь должен хотеть пройти игру до конца.

Фоновая музыка, звуковые эффекты. Музыка на фоне служит для передачи эмоционального настроения. Фоновая музыка больше всего влияет на настроение игрока, а не только заполняет тишину. Звуковые эффекты так же важны, это могут быть шаги по деревянному полу приближающегося врага или звуки приближения автомобиля. Все эти детали необходимы для целостности восприятия игры.

Выпуск продукта. Когда игра собрана, остается только протестировать игру, проверить ее работоспособность и устранить ошибки. Полностью собранная игра, но не проверенная на наличие ошибок называется бета версией. Поиском ошибок занимается тестировщик, бывает, что в качестве тестировщиков выступают обычные игроки.

Последний этап – продажа игры, необходимый элемент для коммерческого проекта.

Поддержка продукта включает в себя обновления игры, а также устранение ошибок, выявленных после запуска в продажу версии [7, с. 13].

Глава 2. Опытнo-экспериментальная работа по созданию игры.

2.1. Результаты анкетирования

Прежде чем приступить к созданию собственной игры и разработке дизайна средствами визуального программирования, мы провели анкетирование, результаты которого представлены ниже. В анкетировании участвовало 93 человека в возрасте 10-12 лет из двух образовательных учреждений МБУ школы №21 г. Тольятти и МБОУДО ДТДМ г.о. Тольятти. По результатам анкетирования среди детей видно, что 91% регулярно играют в компьютерные игры. Процентное соотношение количества играющих и не играющих детей в компьютерные игры отображено на рисунке 1 (Приложение 1).

При этом практические исследования показывают, что дети используют различные устройства для игр. Так, большая часть детей – 52% опрошиваемых играют преимущественно на смартфонах, 35% – отдают предпочтение компьютерам (ноутбукам) и лишь 13% детей используют планшеты. Результаты отображены на рисунке 2 (Приложение 1).

Также в результате опроса мы выяснили какое количество часов в сутки дети посвящают компьютерным играм. Так по утверждению детей 4% опрошиваемых играют 3-4 часа в день, 24% – 2-3 часа в день, 20% – 1-2 часа, при этом большая часть опрошиваемых – 44% играют в день менее часа. В ходе опроса мы учитывали только время, проведенное за компьютерными играми. Время, посвященное просмотру видеороликов и другим развлечениям, не учитывалось. Графическое представление результатов отображено на рисунке 3 (Приложение 1).

Следующий раздел исследования был посвящен интересам детей в области компьютерных игр, а именно каким жанрам отдавали предпочтения ребята. Предпочтения детей распределились следующим образом: большая часть 35% – отметили вариант «квест», на долю ролевых игр и так называемых «бродилок-стрелялок» приходится по 26 % опрошиваемых. Головоломки отметили 17% респондентов, а на тактические RPG, шутеры и логические игры приходится по 9%. Развивающим и музыкальным играм отдают предпочтение лишь 4% опрошиваемых. Процентное соотношение жанров, которые предпочитают дети, отображено на рисунке 4 (Приложение 1).

Также оценивалось отношение родителей к увлечению детей компьютерными играми. По мнению детей, 78% родителей относятся положительно к подобному увлечению и регулярно разрешают детям играть в компьютерные игры. 18% опрошиваемых утверждают, что родители относятся отрицательно к данному увлечению и не разрешают играть в компьютерные игры. 4% опрошиваемых отметили, что не играют в компьютерные игры. Графическое представление результатов отображено на рисунке 5 (Приложение 1).

Одной из важных проблем является зависимость от компьютерных игр и возможность ее развития с раннего возраста. Поэтому мы включили в анкету вопрос о том, насколько сложно прекратить игру. Так, 59% опрошиваемых

утверждают, что без труда могут прекратить игру, в то время как 37% признались, что испытывают затруднения в прекращении компьютерной игры. Процентное соотношение результатов отображено на рисунке 6 (Приложение 1).

На вопрос «Хотели ли бы вы научиться создавать компьютерные игры самостоятельно?» 100% респондентов ответили утвердительно (рисунок 7, Приложение 1). Поэтому было принято решение попробовать разработать дизайн игры самостоятельно и поделиться опытом и результатом работы со сверстниками.

2.2. Критерии выбора и обзор конструкторов для создания игр

Нашей задачей является создание дизайна игры без необходимости изучения сложного синтаксиса языков программирования. Поэтому нам нужно изучить программное обеспечение, с помощью которого можно создавать игры. Интерфейс должен быть на русском языке, простым и интуитивно понятным детям в возрасте 10-12 лет. Кроме этого, приложение должно быть бесплатным и не требовать установки на компьютер или мобильное устройство, чтобы каждый участник исследования мог играть в игру. Также одним из важных моментов является наличие в сети обучающих видео для самостоятельной работы учеников.

Таким образом, мы можем выделить следующие критерии, которым должна отвечать среда визуального программирования:

1. Язык интерфейса – русский
2. Уровень сложности интерфейса – простой
3. Установка на устройство – не требуется
4. Вид лицензии – свободная
5. Наличие обучающих видео – имеется

Изучив справочные материалы, мы выбрали наиболее популярные конструкторы игр: Construct, Kodu Game Lab, GameMaker Studio, Unity, Scratch

Construct 2 – это кроссплатформенный программный продукт для разработки игр. Чтобы работать с ним, не нужны специальные знания о языках программирования и даже какого-либо опыта в игростроении. Интерфейс программы простой и понятный. Есть много видео инструкций по взаимодействию с приложением. Все действия в Construct 2 осуществляются прогрессивным физическим движком, на модели поведения объектов оказывает влияние эффективная система визуализации эпизодов. Можно подключать плагины. В финальном файле сохраненной игры остаются исключительно те модули и алгоритмы поведения, которые используются в этой игре. Поэтому размер конечного продукта выходит небольшим.

Ключевые преимущества программы Construct 2:

- прогрессивная система показа эпизодов;
- адаптивные настройки поведенческих моделей;
- 70 шаблонов эффектов;

- разработка игр для разных операционных систем;
- 30 встроенных модулей с возможностью импортировать или сформировать новые.

Kodu Game Lab – это конструктор для создания 3D игр. Чтобы работать с ним, не нужно разбираться в языках программирования. Справиться с программой смогут даже дети. Основное внимание в ПО уделяется процедуре формирования сюжета и идей игры с использованием логики, не отвлекая постоянными мыслями о методе программирования.

Характеристики приложения:

- визуализирована область создания объектов;
- шаблоны объектов и персонажей с выбором стандартных моделей поведения;
- неограниченные возможности для проектирования;
- предварительно продуманные игровые миры;
- совместимость с Xbox 360 и компьютером;
- можно взаимодействовать с блок-схемами;
- есть версия с поддержкой русского языка.

Хотя приложение и не отличается сложностью в работе с ним, но его создатели добавили к своему проекту и курс обучения. Скорее всего, он будет полезен для детей, которые пробуют свои силы в создании игр.

Gamemaker Studio 2 – уникальная программа для создания маленьких по объемам мультиплатформовых игр. Создатели, пользуясь этим замечательным приложением, могут заниматься творчеством, предоставив программе техническое обеспечение. Удобный интерфейс, отсутствие ручного кодирования привлекает многих. В запасе Gamemaker Studio имеется множество шаблонов и заготовок макетов, которые значительно упрощают работу, необходимо только оговорить правила взаимодействия между собой.

Преимущества:

- работа с множеством платформ одновременно, созданные игры поддерживают различные операционные системы;
- интеграция готовой продукции;
- создание небольших по размерам форматов за короткий промежуток времени;
- по желанию создателя можно изменять функциональные решения.

Unity 3D – мощный программный продукт для разработки компьютерных игр, мобильных девайсов, игровых приставок и Смарт ТВ. Имеет обширное число встроенных скриптов, текстур и макетов. Образование игр происходит на базе игрового движка NVIDIA PhysX. Программу можно скачать бесплатно на русском языке непосредственно на данном сайте. Интерфейс настолько комфортный и простой, что не составит труда понять суть всех инструментов и возможностей и на английском.

Ключевые характеристики Unity 3D:

- конструктор всецело объединен со средой разработки;

- совместимость с DirectX и OpenGL;
- просто добавлять текстуры (в каком-либо известном графическом формате), модели и звуки;
- создание скриптов на языке JavaScript и написание алгоритмов на C#;
- много возможностей настройки;
- присутствие функции Drag-and-drop для перемещения объектов;
- встроенная совместимость с сетью;
- применение Unity Web Player для открытия игры через браузер.

Используя программу можно разрабатывать компьютерные игры в каком угодно жанре. Создатель игры может без особых усилий добавить желаемую текстуру, макет, звук. Можно импортировать текстуры всех известных графических форматов изображений.

Scratch – визуально-ориентированная среда программирования, которая и сама похожа на игру. В ней не требуется писать сложных кодов и скриптов, достаточно работать в ней, как в редакторе, добавляя объекты и определенные команды. Scratch – программа предназначена для детей от восьми лет для изучения основ программирования и математики. Автор создал платформу, где в формате игры можно донести до школьника математические законы и начальные позиции составления компьютерных программ и дизайна. Интерфейс программы Scratch простой, удобный и, конечно же, красочный, поскольку ориентирован на детскую аудиторию. Навигация очень подробная и комфортная, интуитивно понятное меню. Утилита представляет собой целую платформу для разработки интересных проектов. Игры и мультики, созданные собственноручно, можно выложить в общую платформу, для оценивания другими пользователями. Одновременно Scratch дает возможность позволяет рассматривать проекты других пользователей, оценивать их, комментировать и дорабатывать. Рабочая зона платформы делится на три части:

- первая – это разработка своего сценария;
- вторая – блоковая система, то есть программа проекта;
- третья – результат выделенного скрипта.

В зоне сценария доступны функции звука, пера, движения, внешнего вида, сенсоров, переменных, чисел и управления. Каждая функция имеет подсказки. Можно установить среду визуального программирования на компьютер или использовать онлайн, работая в браузере.

Сравнительная таблица характеристик конструкторов компьютерных игр в соответствии с разработанными критериями представлена в таблице (Приложение 2)

Проанализировав характеристики наиболее популярных конструкторов в соответствии с разработанными нами критерии, мы пришли к выводу, что среда визуального программирования Scratch отвечает всем нашим требованиям. Рассмотрим более подробно интерфейс среды визуального программирования Scratch.

2.3. Интерфейс и основы работы в среде визуального программирования

После запуска программы или сайта перед нами появляется окно, в котором можно выделить три части (три столбца). В свою очередь каждый столбец разделен на две области: верхнюю и нижнюю. Все ячейки разные и каждая из них предназначена для своей конкретной цели. Интерфейс среды визуального программирования Scratch представлен на рисунке 8 (Приложение 3) [3, с. 14].

Верхняя ячейка в первом столбце содержит восемь кнопок, которые отвечают за движение, контроль, внешность, сенсоры, звук, операторы, перо, переменные. При включении одной кнопки все остальные выключаются. Включенная кнопка вся окрашивается в соответствующий ей цвет. При этом в нижней ячейке первого столбца появляются команды, связанные с включенной кнопкой. У всех кнопок разные привязанные к ним команды. Категории, содержащие блоки для создания скриптов представлены на рисунке 9 (Приложение 3).

В верхней ячейке второго столбца показаны свойства объекта, которым мы будем управлять (создавать программы для него). Область настройки параметров спрайта (вверху) и область списка спрайтов (внизу) представлена на рисунке 10 (Приложение 3). По умолчанию – это кот. Его имя написано в поле – Спрайт1. Спрайтами в Scratch называются все объекты по умолчанию, отличаются лишь их номера. Обычно предполагается, что мы сами их будем переименовывать и давать им имена [8, с. 19].

Ниже поля имени отображены три свойства объекта (в данном случае, кота) – это его положение (координаты x и y) и направление.

Внизу мы видим три кнопки-вкладки – скрипты, костюмы и звуки. От того какая из них нажата, зависит ячейка внизу. Если нажата кнопка скрипты, то нижняя ячейка второго столбца покажет программы (скрипты) для объекта, который отображен в верхней ячейке. Кнопки костюмы и звуки позволяют настраивать и менять соответственно внешний вид объекта и издаваемые им звуки.

В третьем столбце в верхней ячейке мы видим белое поле и кота на нем. Белое поле – это холст. Многие действия, которые задает программист на вкладке скрипты, объект выполняет именно на холсте. Так, если мы запрограммируем ходьбу кота, то он будет перемещаться по холсту.

Если зажать левую кнопку мыши над котом, а затем, не отпуская ее, перемещать мышью, то кот будет перемещаться. Таким образом, мы можем менять его положение на холсте.

Над холстом находятся две кнопки для управления игрой. Когда для объектов (кота и других) будут составлены какие-либо скрипты (программы) на вкладке скрипты, то чтобы объекты начали их выполнять, надо нажать зеленый флажок. Чтобы остановить выполнение, надо нажать красный круг. Кнопки запуска и остановки игры представлены на рисунке 11 (Приложение 3).

Справа от кнопок запуска и остановки игры находятся еще три кнопки, которые предназначены для изменения всего окна приложения Scratch. Кнопки

для управления внешним видом рабочей области среды визуального программирования Scratch представлены на рисунке 12 (Приложение 3).

2.4. Разработка платформера

Результаты исследования показали – большинство опрошенных отметили, что наиболее интересным для них являются игры в жанре квест и так называемые «бродилки». Квест – это компьютерная игра, сюжет которой построен на путешествии к конкретной цели. В квесте нужно решить цепочку логических загадок. Главный герой в квесте управляется игроком. Так как наша цель состоит в том, чтобы показать, что игру может создать каждый ученик класса, то игра, которую мы будем создавать должна быть достаточно проста в исполнении. Поэтому мы решили сделать платформер, который близок по содержанию к квесту.

Описание правил игры и контент-дизайн. Название игры «Конфетный платформер». Цель игры – добраться до финиша, чтобы победить. Игра рассчитана на одного игрока и имеет 25 небольших уровней.

Сюжет. Главный герой по имени «Чупсик» путешествует по конфетному миру, преодолевая препятствия. На его пути попадает опасная лава, которую надо избегать. Если главный герой касается лавы, то уровень перезагружается. В процессе прохождения игры главный герой может воспользоваться лифтом и батутом, расположенным на некоторых уровнях.

Управление. Управление главным героем осуществляется с помощью клавиатуры клавишами «Стрелки».

Дизайн уровней.

Процесс разработки дизайна игры в среде визуального программирования Scratch проходил в несколько этапов:

1. Подготовка изображений спрайтов и фонов.
2. Создание скриптов.
3. Добавление звуков.
4. Тестирование игры.
5. Устранение ошибок.

На этапе подготовки изображений необходимо было найти в сети Интернет картинки по выбранной нами теме. Часть изображений были нарисованы и отредактированы во встроенном графическом редакторе Scratch. Были подготовлены следующие изображения: фоны для смены уровней, изображение главного героя, изображение для сообщения о победе. Изображения были загружены в среду визуального программирования в виде спрайтов и костюмов для них. Также были добавлены объекты, с которыми главный герой взаимодействует в процессе игры – платформы, батуты, лифты и телепорт, для перехода на следующий уровень.

На следующем этапе были разработаны скрипты – программы для спрайтов. В программу главного героя заложен алгоритм, который позволяет управлять его движением и сменой костюма клавишами клавиатуры. Созданы скрипты для взаимодействия с объектами на уровнях, а также смена сцены для следующего уровня или перезагрузки уровня. Скрипты для управления героем,

взаимодействия с объектами и сменой уровней представлены на рисунках 37 – 47 (Приложение 5).

Звуки были добавлены из библиотеки звуков среды визуального программирования и открытых источников сети Интернет. Скрипты для создания звуков в игре отображены на рисунках 47 – 49 (Приложение 3).

После создания спрайтов, написания скриптов, игра была протестирована и выявлены ошибки. После тестирования игры выяснилось, что батут, который должен подкидывать главного героя работает некорректно. На этапе исправления ошибок этот недочет был устранен.

Заключение

При выполнении данной исследовательской работы мы выяснили:

1. Компьютерные игры являются очень популярными. Особенно активно развивается сектор мобильной разработки.

2. Ежегодно количество людей, увлекающихся компьютерными играми растет.

3. В сети Интернет появляются не только коммерческие проекты, но и проекты разработчиков-любителей. Пользователи имеют свободный доступ к конструкторам игр и средам визуального программирования, которые позволяют создавать игры без знания специальных языков программирования.

4. Платформы для создания игр доступны для разных возрастных категорий. Существуют конструкторы, которые могут освоить учащиеся в возрасте 10-12 лет и младше. Изучив видеоуроки, можно самостоятельно создать дизайн компьютерной игры, не устанавливая приложения на компьютер.

5. В ходе работы, проанализировав конструкторы для создания игр, самостоятельно создав несколько игр, мы можем утверждать, что поставленная нами цель достигнута, гипотеза подтвердилась: дети 10-12 лет могут самостоятельно разрабатывать компьютерные игры, не используя для этого сложных языков программирования.

Список литературы:

1. Васильев А. А. Термин «Компьютерная игра»: опыт междисциплинарного анализа / А. А. Васильев, Ю. В. Печатнова // Пролог: журнал о праве. – 2021. – №2 (30). – С. 131–138.
2. Галузин А. Создание игрового уровня за 11 дней/ А. Галузин. – Интернет-издание, 2010 – 44 с.
3. Голиков Д. В. Scratch для юных программистов / Д.В. Голиков – СПб.: БХВ-Петербург, 2017. – 192 с.: ил.
4. Казакова Н.Ю. Парадигма развития гейм-дизайна как вида проектной деятельности на основании этапов развития аппаратной части/ Н.Ю. Казакова// Вестник Кемеровского государственного университета культуры и искусств. –2016. – №35. – С. 134–145
5. Калимулина О. В. Особенности рынка компьютерных игр/ О. В. Калимулина // Российский внешнеэкономический вестник. – 2010. – №5. – С. 45 – 51.
6. Костер Р. Разработка и теория развлечений /Р. Костер. пер. с англ. О.В. Готлиб. – М.: ДМК Пресс, 2018. – 288 с.
7. Лазарева О. Ю. Основные этапы создания видеоигр / О. Ю. Лазарева, А. В. Санина // Молодой ученый. – 2019. – № 4 (242). – С. 12-14.
8. Маржи М. Scratch для детей. Самоучитель по программированию / М. Маржи, пер. с англ. М. Гескиной и С. Таскаевой – М.: Манн, Иванов и Фербер, 2017. – 288 с.
9. Федоров А. В. Краткая история компьютерных игр в России/ А. В. Федоров // Медиаобразование. – 2013. – №4. – С. 13 -149.
10. Шелл Д. Геймдизайн. Как создать игру, в которую будут играть все / Д. Шелл – «Альпина Диджитал», 2015. – 640 с.

Анализ результатов анкетирования



Рисунок 1 – Количество детей в классе, которые играют в компьютерные игры (%)

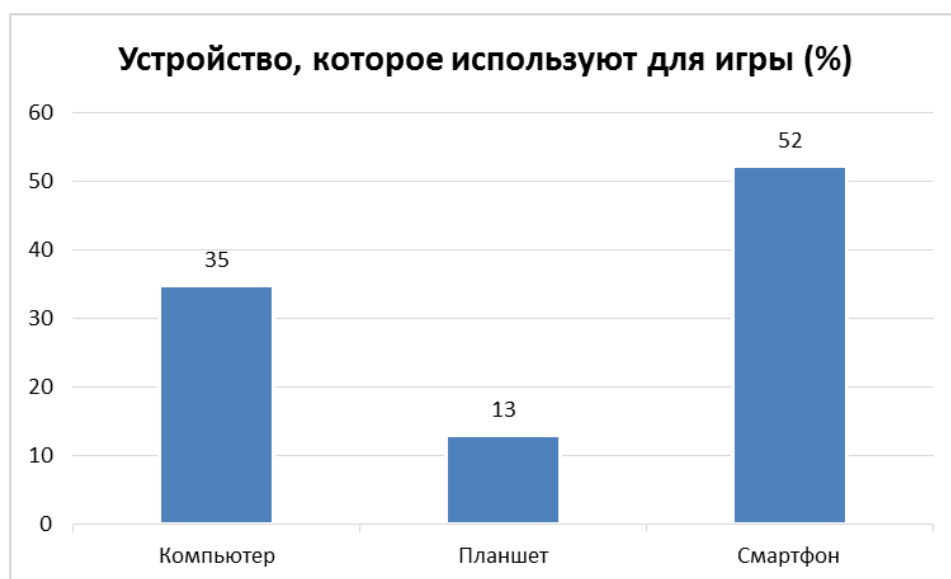


Рисунок 2 – Устройство которое дети используют для компьютерных игр (%)

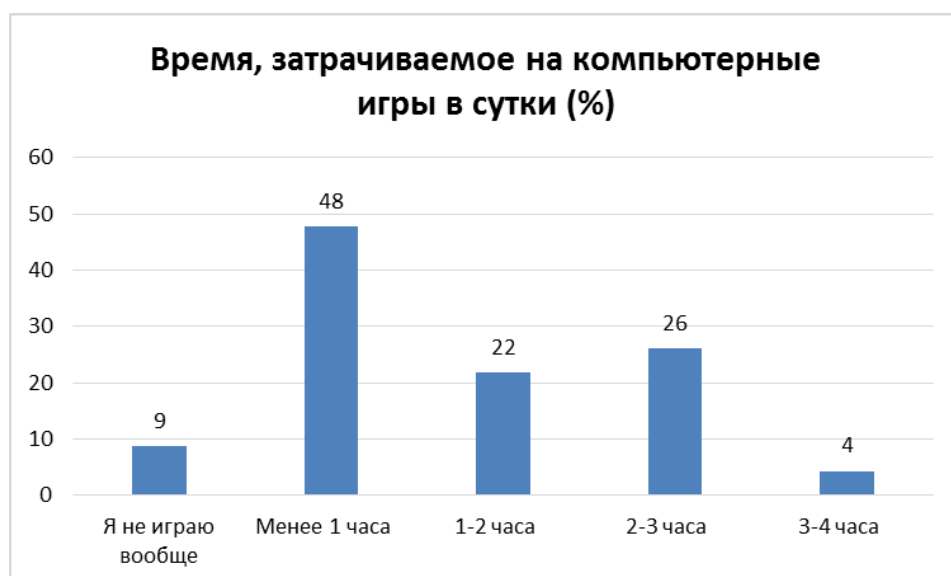


Рисунок 3 – Время, которое дети затрачивают на компьютерные игры в сутки (%)



Рисунок 4 – Жанры игр, которые предпочитают дети (%)



Рисунок 5 – Разрешают ли родители играть в компьютерные игры (%)



Рисунок 6 – Затруднения в прекращении игры (%)



Рисунок 7 – Желание научиться создавать компьютерные игры самостоятельно

Сравнительная таблица характеристик конструкторов компьютерных игр в соответствии с критериями

№ п/п	Название	Язык интерфейса	Уровень сложности интерфейса	Установка на устройство	Вид лицензии	Наличие обучающих видео
1	Construct	Английский	Средний	Требуется	Платная	Имеются
3	Scratch	Русский	Простой	Не требуется	Бесплатная	Имеются
4	Unity	Английский	Сложный	Требуется	Бесплатная	Имеются
2	GameMaker Studio	Русский	Средний	Требуется	Платная	Имеются
5	Kodu Game Lab	Русский	Простой	Требуется	Бесплатная	Имеются

Элементы интерфейса среды визуального программирования

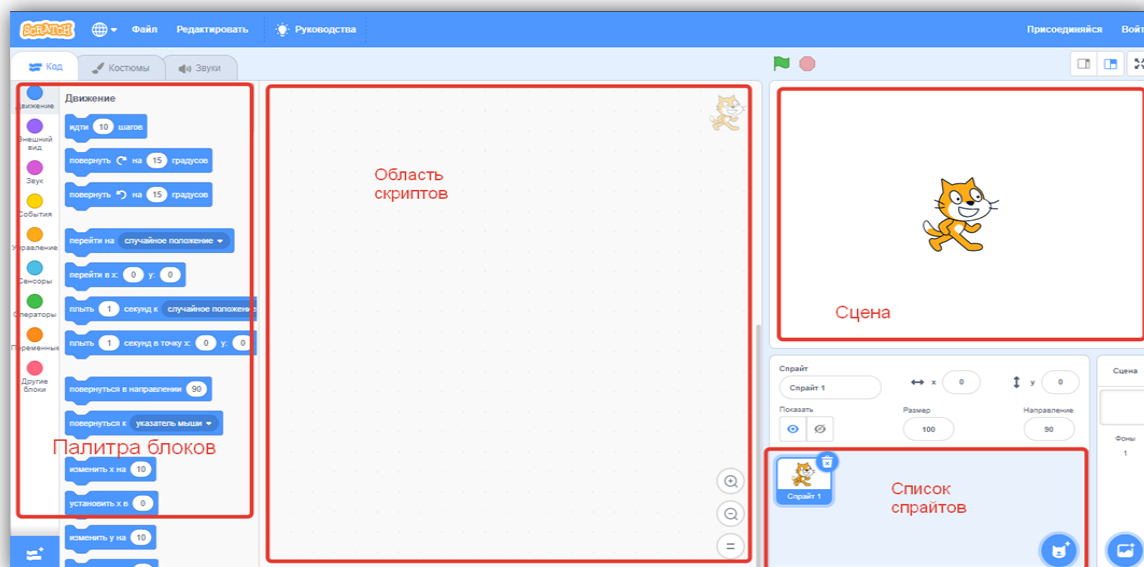


Рисунок 8 – Интерфейс среды визуального программирования Scratch

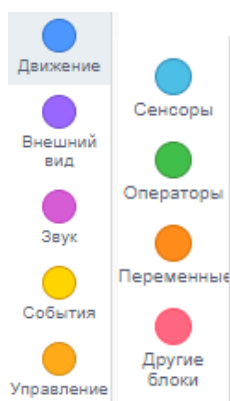


Рисунок 9 – Категории, содержащие блоки для создания скриптов

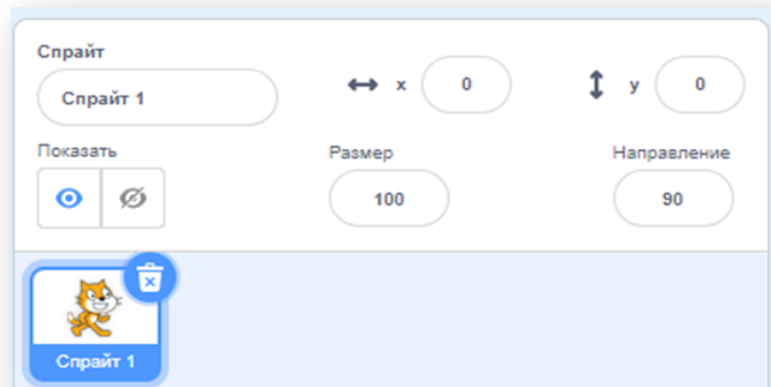


Рисунок 10 – Область настройки параметров спрайта (вверху) и область списка спрайтов (внизу)



Рисунок 11 – Кнопки запуска и остановки игры



Рисунок 12 – Кнопки для управления внешним видом рабочей области среды визуального программирования Scratch

Дизайн уровней

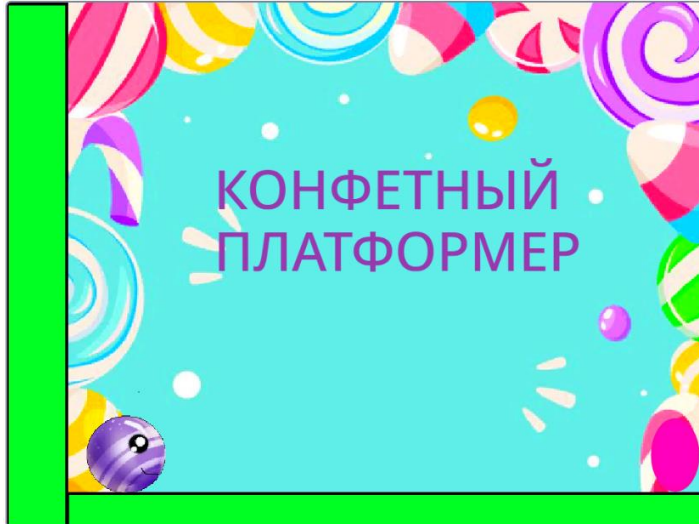


Рисунок 13 – Дизайн уровня 1

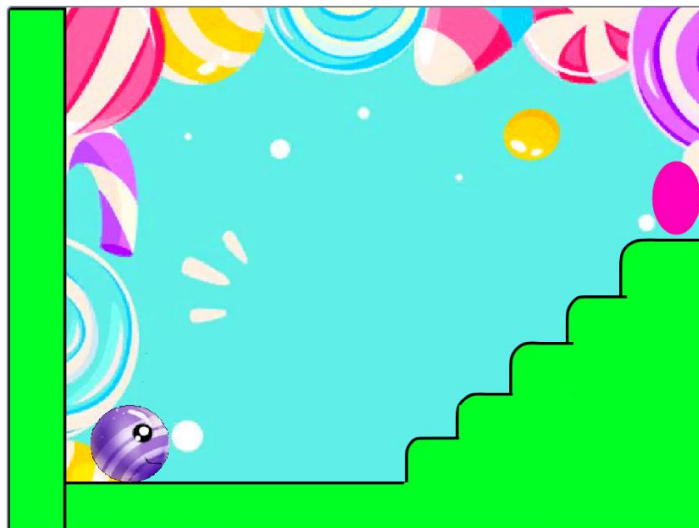


Рисунок 14 – Дизайн уровня 2



Рисунок 15 – Дизайн уровня 3

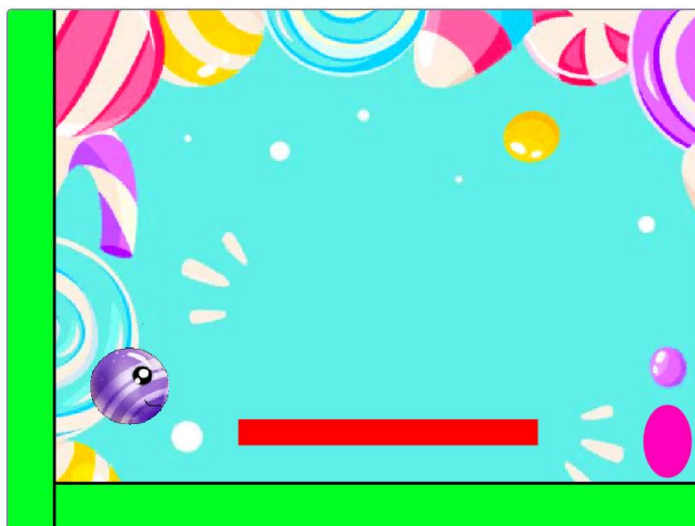


Рисунок 16 – Дизайн уровня 4



Рисунок 17 – Дизайн уровня 5

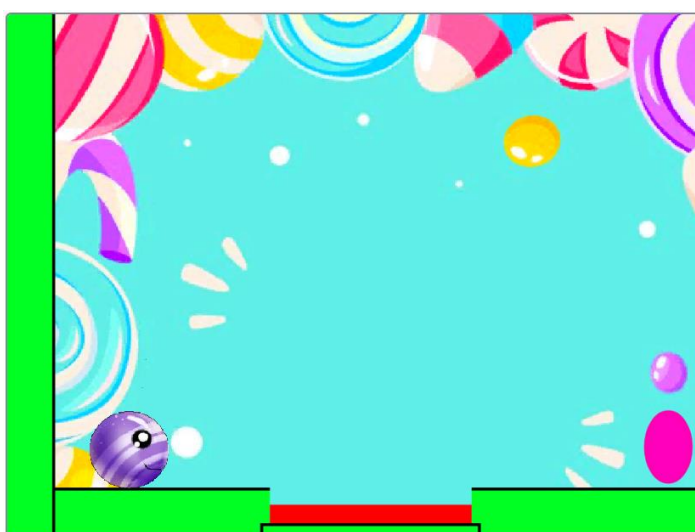


Рисунок 18 – Дизайн уровня 6



Рисунок 19 – Дизайн уровня 7



Рисунок 20 – Дизайн уровня 8



Рисунок 21 – Дизайн уровня 9



Рисунок 22 – Дизайн уровня 10



Рисунок 23 – Дизайн уровня 11

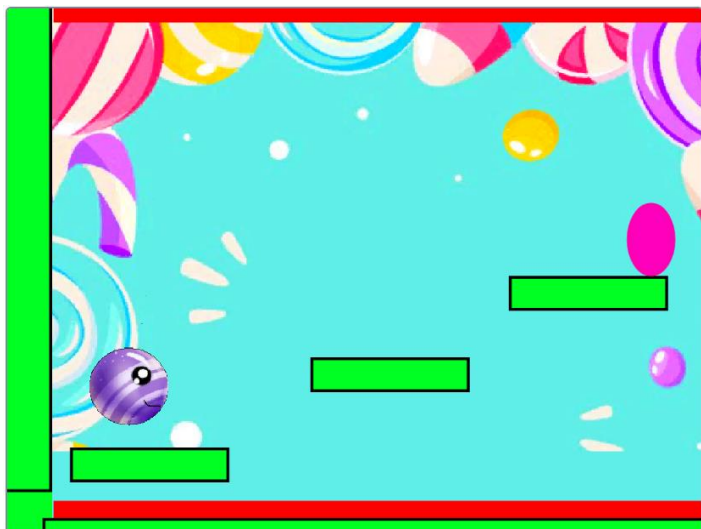


Рисунок 24 – Дизайн уровня 12



Рисунок 25 – Дизайн уровня 13



Рисунок 26 – Дизайн уровня 14



Рисунок 27 – Дизайн уровня 15



Рисунок 28 – Дизайн уровня 16



Рисунок 29 – Дизайн уровня 17



Рисунок 30 – Дизайн уровня 18



Рисунок 31 – Дизайн уровня 19

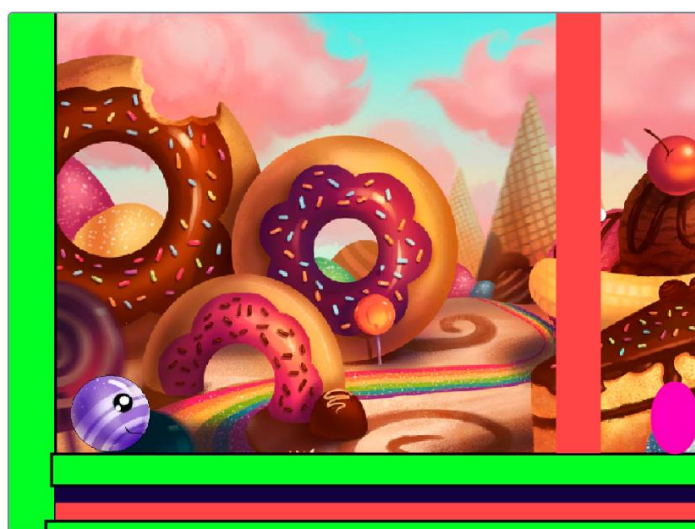


Рисунок 32 – Дизайн уровня 20



Рисунок 33 – Дизайн уровня 21

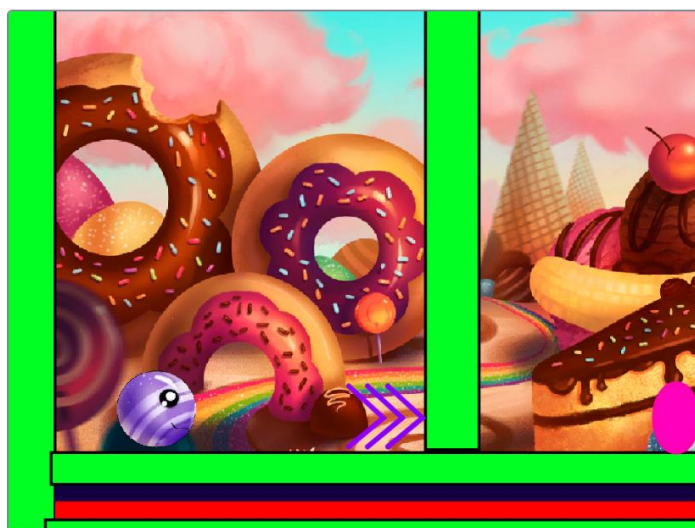


Рисунок 34 – Дизайн уровня 22



Рисунок 35 – Дизайн уровня 23



Рисунок 36 – Дизайн уровня 24

Разработка скриптов

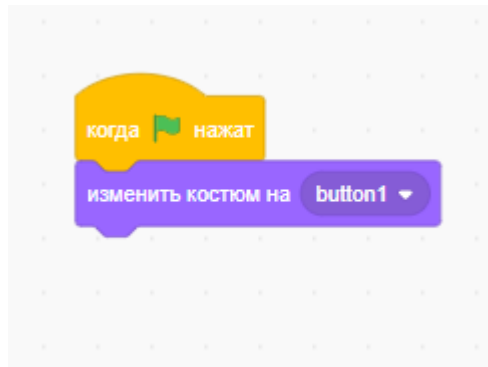


Рисунок 37 – Скрипт для установки костюма при запуске игры

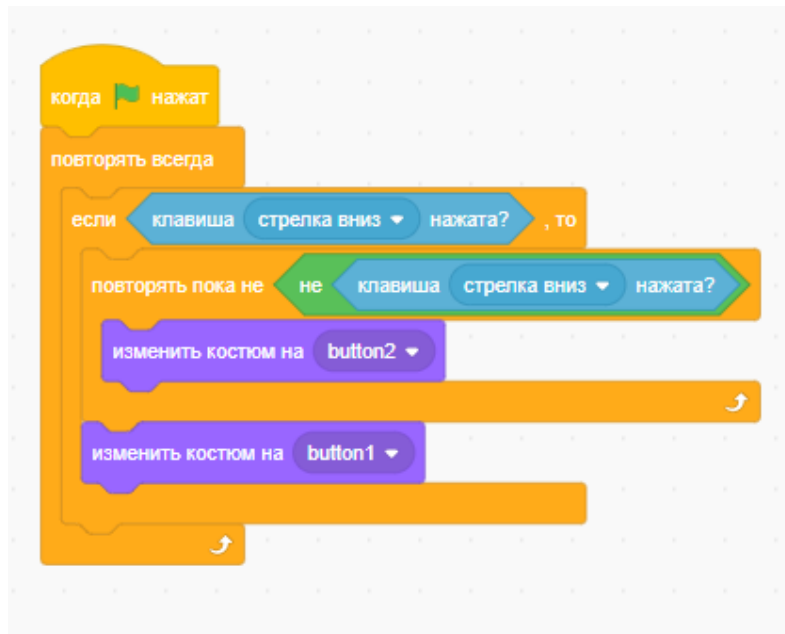


Рисунок 38 – Скрипт для смены костюма при нажатой клавише «стрелка вниз»

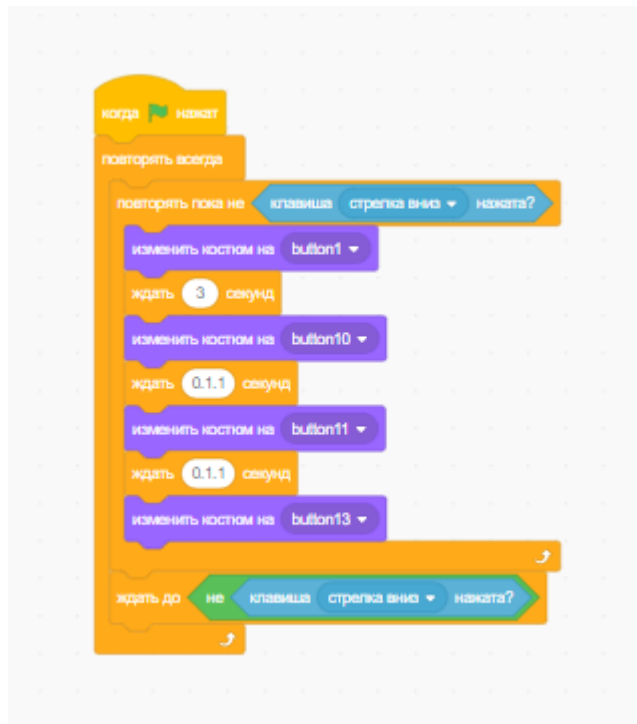


Рисунок 39 – Скрипт для смены костюма во время ожидания

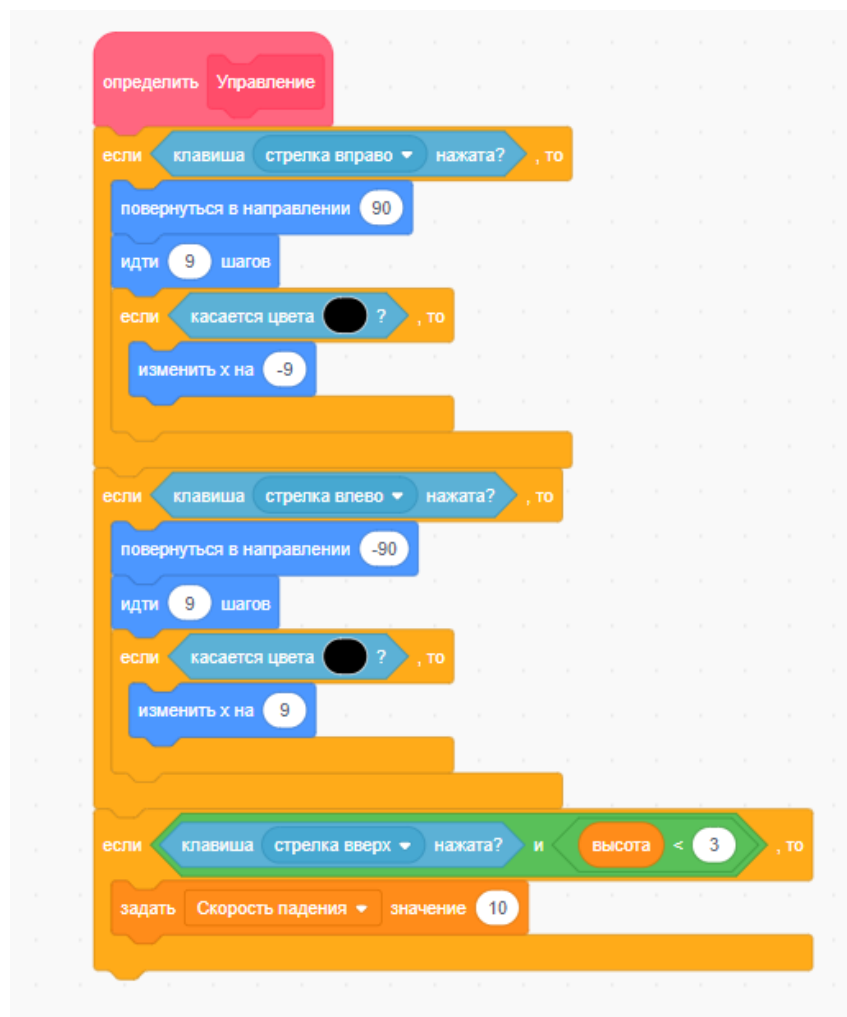


Рисунок 40 – Скрипт для управления главным героем и взаимодействия с поверхностью

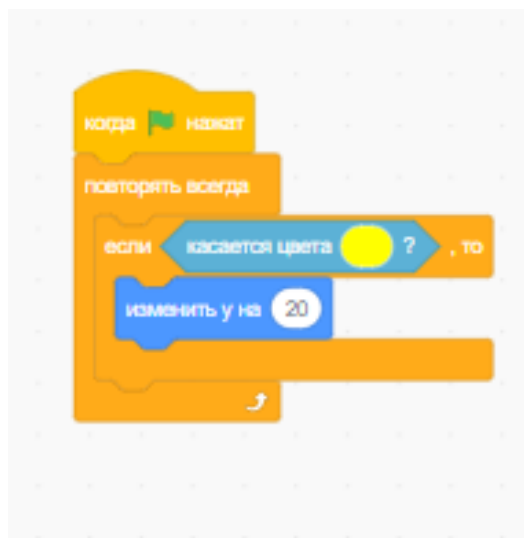


Рисунок 41 – Скрипт для взаимодействия с лифтом

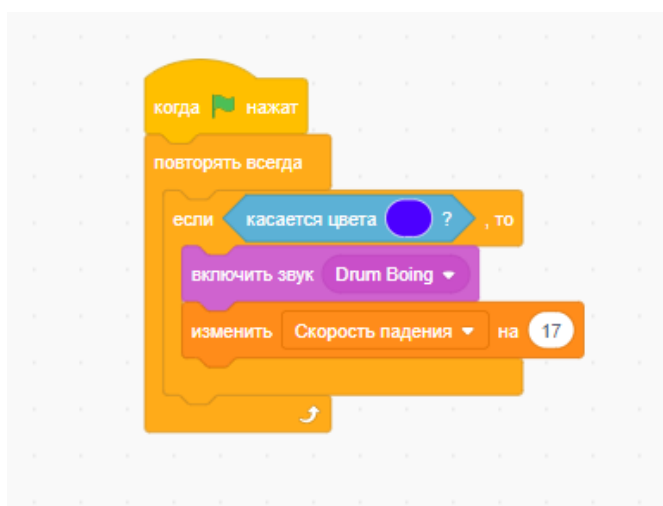


Рисунок 42 – Скрипт для взаимодействия с батутом

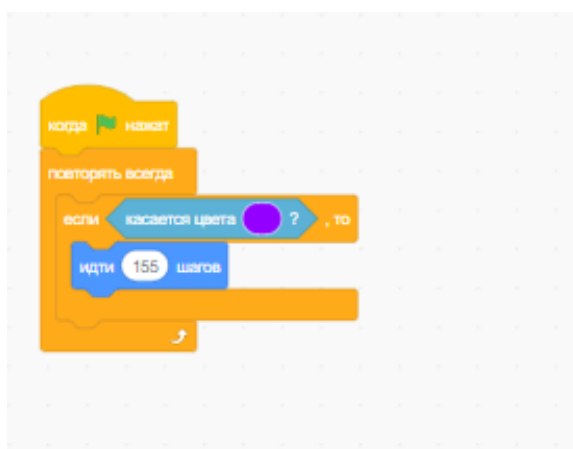


Рисунок 43 – Скрипт для взаимодействия с порталом и перехода на следующий уровень

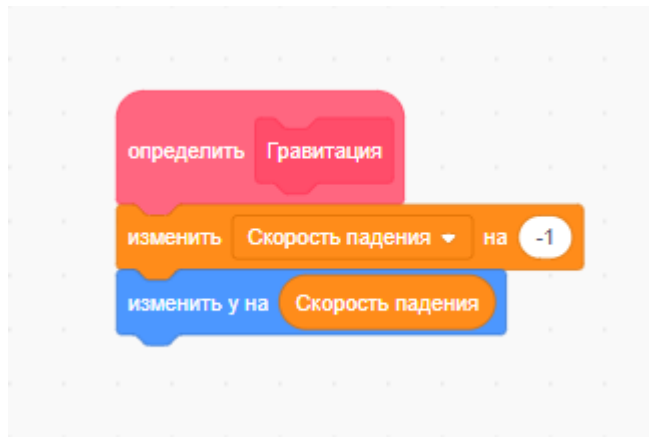


Рисунок 44 – Скрипт для появления главного героя на новом уровне

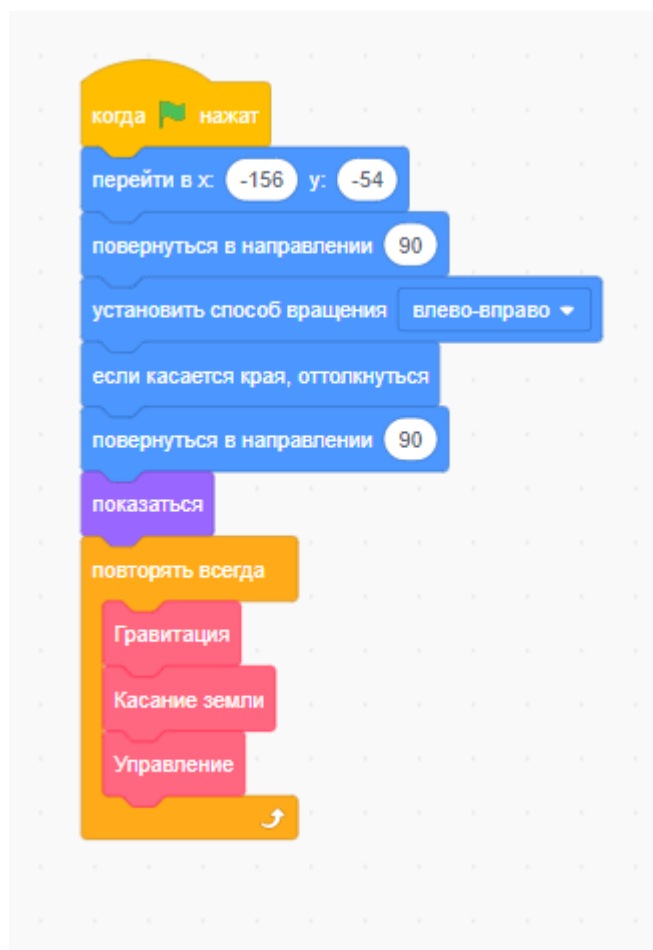


Рисунок 45 – Скрипт для запуска основных команд, определяющих поведение героя в пространстве

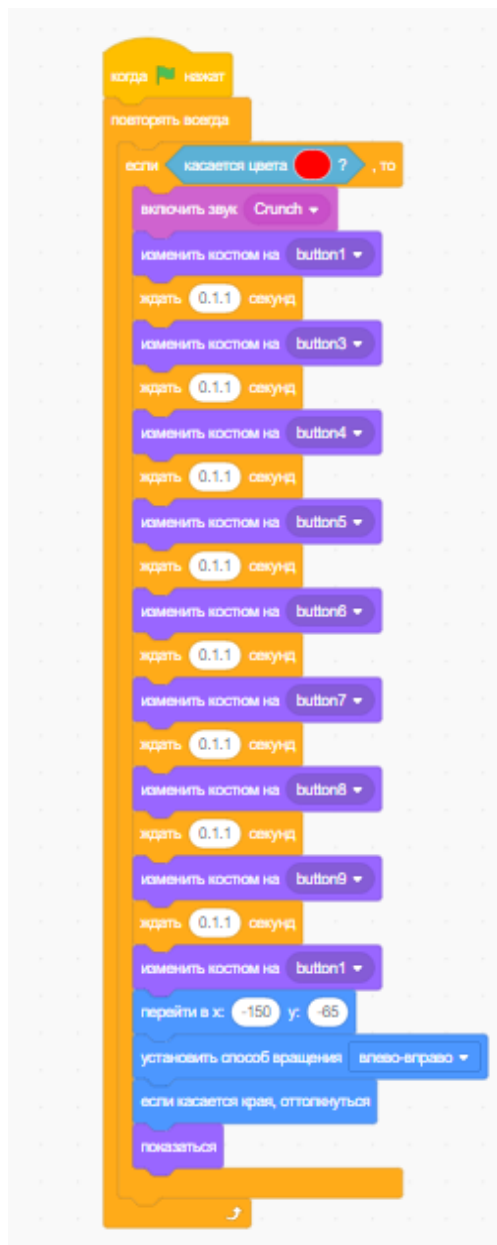


Рисунок 46 – Скрипт для взаимодействия с лавой



Рисунок 47 – Скрипт для взаимодействия с поверхностью

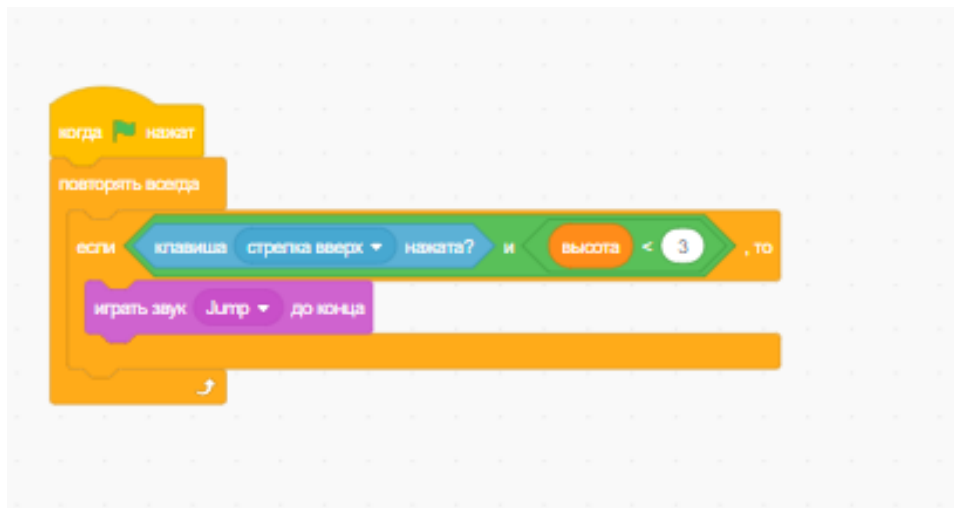


Рисунок 48 – Скрипт для создания звука во время прыжка



Рисунок 49 – Скрипт для создания звука во время перехода на новый уровень