

Е. Г. Литвак, Н. В. Брадул

победители Всероссийского конкурса «Дистанционное обучение: новый опыт в новой реальности»
в номинации «Дистанционное обучение: новый опыт в новой реальности»,
Донецкая академия управления и государственной службы, г. Донецк, Украина

ОПЫТ ОРГАНИЗАЦИИ КУРСА «ОСНОВЫ ПРОГРАММИРОВАНИЯ ДЛЯ ШКОЛЬНИКОВ» ЦЕНТРОМ ДОВУЗОВСКОЙ ПОДГОТОВКИ В УСЛОВИЯХ ПАНДЕМИИ

Аннотация

В статье рассмотрены процессы проектирования, разработки и запуска онлайн-курса «Основы программирования для школьников» для центра довузовской подготовки в условиях пандемии. Обоснован выбор верхнеуровневой модели проектирования SAM1 для курсов, которые необходимо разработать и запустить в сжатые сроки, не имея полных данных о целевой аудитории. Показаны преимущества выбора модели «обратного» проектирования для разработки содержания целевого курса. С помощью метода «обратного» проектирования спроектирован и разработан курс «Основы программирования для школьников». Представлена программа курса в виде последовательности модулей и отдельных тем. Обосновано использование элементов геймификации на базе социальной сети «ВКонтакте». Приведены примеры учебных активностей. Даны ссылка на разработанный курс, размещенный в LMS Moodle Донецкой академии управления и государственной службы.

Ключевые слова: программирование, проектирование образовательной программы, JavaScript, онлайн-формат обучения, модель «обратного» проектирования.

DOI: 10.32517/2221-1993-2021-20-5-5-12

1. Введение

Центр довузовской подготовки (ЦДП) является базовой профориентационной структурой Донецкой академии управления и государственной службы. После прохождения обучения в ЦДП ежегодно до 80 % выпускников становятся студентами академии, что составляет 20–25 % набора первого курса.

Деятельность ЦДП направлена, прежде всего, на повышение качества знаний выпускников школ и организаций СПО, проведение профориентационной работы по поиску и привлечению в академию одаренной молодежи, предоставление слушателям первичных знаний и практических навыков по тем направлениям подготовки, прием на которые открыт в академии.

Одним из популярных и востребованных направлений подготовки является направление 09.03.03 «При-

кладная информатика». Для подготовки школьников к поступлению на это направление помимо базовой программы по математике в ЦДП существует «Школа юного программиста», где старшеклассники могут получить углубленные знания по основам программирования, а также более детальное представление о работе в сфере информационных технологий.

Следует отметить, что, несмотря на наличие в школьной программе по информатике тем, связанных с алгоритмизацией и программированием, школьники редко имеют в этой области полноценный объем знаний, предусмотренный школьной программой. Это объясняется следующими причинами:

1. Алгоритмизация и программирование — это сферы деятельности, требующие принципиально новых навыков, которые не вырабатываются в процессе изучения других школьных предметов,

Контактная информация

Литвак Елена Геннадиевна, канд. экон. наук, старший преподаватель кафедры информационных технологий, Донецкая академия управления и государственной службы, г. Донецк, Украина; адрес: 83015, Украина, г. Донецк, ул. Челюскинцев, д. 163а; e-mail: alttt@yandex.ru

Брадул Наталья Валерьевна, канд. физ.-мат. наук, зав. кафедрой информационных технологий, Донецкая академия управления и государственной службы, г. Донецк, Украина; адрес: 83015, Украина, г. Донецк, ул. Челюскинцев, д. 163а; e-mail: bradnv@ukr.net

E. G. Litvak, N. V. Bradul

Donetsk Academy of Management and Public Administration, Donetsk, Ukraine

EXPERIENCE OF ORGANIZATION OF THE COURSE "THE BASICS OF PROGRAMMING FOR SCHOOLCHILDREN" BY THE CENTER OF PRE-UNIVERSITY TRAINING IN A PANDEMIC

Abstract

The article discusses the processes of designing, developing and launching an online course "The basics of programming for schoolchildren" for the center of pre-university training in a pandemic. The choice of the top-level design model SAM1 for courses that need to be developed and launched in a short time without complete data on the target audience is substantiated. The advantages of choosing a backward design model for developing the content of the target course are shown. Using the method of backward design, the course "The basics of programming for schoolchildren" was designed and developed. The program of the course is presented in the form of a sequence of modules and individual topics. The use of gamification elements based on the "VKontakte" social network has been substantiated. Examples of educational activities are given. A link is given to the developed course posted in LMS Moodle Donetsk Academy of Management and Public Administration.

Keywords: programming, coding, designing an educational program, JavaScript, online training format, backward design.

- поэтому учащимся бывает трудно преодолеть некоторый «входной порог».
2. Не все школы располагают достаточно квалифицированным кадровым составом учителей информатики, поэтому программа зачастую упрощается самим учителем, а оценка школьным педагогом уровня знаний его учеников оказывается завышенной.
 3. Разрыв между тем, что школьники видят в компьютерных играх, и тем, что нужно изучать на уроках информатики, очень велик. В играх графические персонажи выглядят и двигаются, почти как живые актеры, а на уроках информатики необходимо складывать числа и искать площадь прямоугольника, что вызывает некоторое разочарование.
 4. Родители могут помочь только в том случае, если информационные технологии — это сфера их профессиональной деятельности. Если же они не используют ИТ в своей профессии, то, скорее всего, они сами испытывали в школе те же трудности.

Сталкиваясь с разочарованием, непониманием и невозможностью преодолеть возникающие трудности в освоении программирования, школьники перестают рассматривать сферу ИТ как область своей будущей профессиональной деятельности.

По вышеперечисленным причинам ЦДП ставит своей целью помочь школьникам преодолеть «входной порог» в область ИТ и пробудить у ребят интерес к программированию.

До пандемии программа обучения в «Школе юного программиста» строилась на основе платформы Unity 2D и языка C# и была рассчитана на очное посещение занятий школьниками. Такое содержание курса было выбрано потому, что, с одной стороны, оно давало представление о языке программирования и принципах составления алгоритмов, с другой — яркая визуализация и ориентация на программирование игр поддерживали необходимый уровень интереса у учащихся.

С принятием карантинных мер в апреле 2020 года ведение занятий по привычной программе стало проблематичным. Находясь в аудитории за компьютерами, на которые корректно установлено нужное программное обеспечение, при непосредственном контакте с преподавателем учащиеся, как правило, не испытывали трудностей с освоением программы обучения. Но, имея в распоряжении домашний компьютер или ноутбук, не всякий школьник самостоятельно справится с установкой необходимого ПО, настройкой дополнительных компонентов, которые могут потребоваться в процессе обучения, а также обеспечением их корректной работы. Поэтому перенос существующего курса в онлайн-режим сразу же столкнулся с препятствиями. Кроме того, на фоне ухудшения общей психологической обстановки в связи с пандемией, при неготовности школьников к регулярным самостоятельным занятиям дома учащиеся быстро теряли интерес и мотивацию.

Для того чтобы преодолеть эти трудности, потребовался курс в формате онлайн, спроектированный с учетом принципиально новых требований. А именно:

1. Курс по-прежнему должен давать школьникам базовые знания по программированию и алгоритмизации.

2. Курс не должен требовать установки ПО, которое может нуждаться в дополнительных компонентах и некорректно работать на некоторых операционных системах.
3. В идеальном случае требуемое для прохождения курса ПО должно быть кроссплатформенным, так как во многих организациях РФ начинается масштабный переход на операционные системы семейства Linux и через некоторое время Linux может стать популярным и для домашних компьютеров.
4. Курс должен быть построен так, чтобы учащиеся как можно быстрее увидели результат своей работы, который должен быть красочным и впечатляющим. Это позволит удерживать интерес и сохранять мотивацию в процессе онлайн-обучения. Поэтому желательно, чтобы в курсе использовались графика и анимация.
5. Курс должен содержать элементы геймификации, которые также послужат фактором сохранения мотивации к прохождению его до конца.

2. Обоснование выбора верхнеуровневой модели проектирования курса

Для проектирования образовательных программ и обучающих курсов обычно используются верхнеуровневые модели проектирования. Наиболее полно и формально весь цикл проектирования представлен моделью ADDIE [11, с. 11]. Модель ADDIE (аббревиатура от англ. Analysis, Design, Development, Implement, Evaluate) подробно описывает пять стадий создания курса:

1. Анализ (Analysis).
2. Создание проекта курса (Design).
3. Разработка курса (Development).
4. Запуск курса (Implement).
5. Оценка (Evaluate).

При этом каждая из пяти стадий начинается после окончания предыдущей, что делает модель ADDIE несколько сходной с «водопадной» моделью разработки ПО.

Однако в условиях внезапного наступления карантина запуск курса должен был состояться в сжатые сроки. При этом правильность выбора контента, формы подачи и видов проведения занятий можно было оценить только после проведения первых нескольких занятий в связи с отсутствием обратной связи с аудиторией. Поэтому модель ADDIE невозможно было использовать в соответствующих ограничениях.

По перечисленным выше причинам в качестве верхнеуровневой модели была выбрана модель **SAM1 (Successive Approximation Model — модель последовательного приближения)**, SAM1 — базовый упрощенный вариант, SAM2 или SAM — расширенный вариант), которая, по мнению ряда авторов, имеет много общего с итеративными методиками разработки ПО (например, Scrum) [10, с. 1].

Модель SAM1 включает три стадии: анализ, разработка и оценка. Эти стадии циклически сменяют друг друга (рис. 1). Таким образом, курс разрабатывается итеративно и каждая следующая стадия анализа основана на оценивании результатов предыдущей итерации. Это позволяет осуществить запуск первого модуля курса, получить обратную связь от учащихся, провести оценку

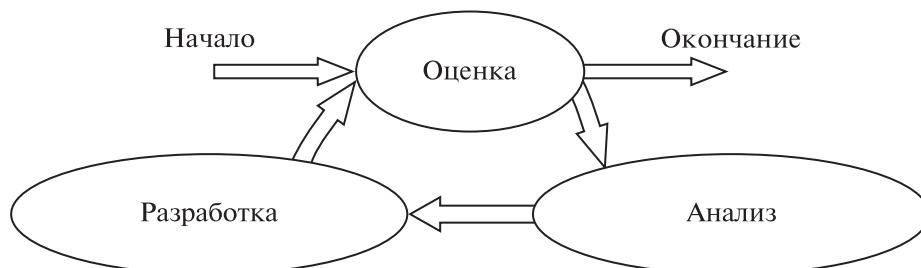


Рис. 1. Схематичное изображение верхнеуровневой модели проектирования SAMI

своих действий и учесть полученную информацию на следующей стадии анализа при проектировании следующих модулей. Таким образом, запуск курса можно осуществить в сжатые сроки, постепенно совершенствуя и адаптируя его под аудиторию.

Следует отметить, что после завершения всего курса, имея в распоряжении полные данные о результатах его прохождения, возникших трудностях и проблемах, можно проектировать новую версию этого же курса для следующего учебного года на основе традиционной модели ADDIE, так как уже имеются данные для глубокого анализа аудитории, достаточно время для выполнения всего цикла проектирования и рабочий прототип курса.

3. Обоснование выбора модели проектирования содержания курса

Специалисты различают три основных модели проектирования содержания курса. В англоязычной литературе они называются 1) Forward Design, 2) Central Design и 3) Backward Design. Наиболее точный перевод на русский язык будет звучать как:

- 1) «прямое» проектирование — от контента;
- 2) проектирование от формы проведения занятий;
- 3) «обратное» проектирование — от целей курса.

Наиболее глубокий сравнительный анализ этих моделей проектирования дан в работе J. C. Richards [13], в которой автор рассматривает перечисленные модели в применении к области филологии. Однако для других областей знания этот сравнительный анализ также справедлив.

В работе J. C. Richards показано, что большинство педагогов, работающих в образовательных учреждениях (как в школах, так и в университетах), привыкли не задумываясь использовать в своей профессиональной деятельности «прямое» проектирование. Такой подход подразумевает, что преподаватель заранее определяет контент, который желает представить в курсе. Формат курса, формы учебных активностей и система оценивания выбираются исходя из контента.

Проектирование курса от формы занятий чаще всего используется в работе с обучающимися определенных возрастных категорий (например, с младшими школьниками), для которых форма подачи информации является приоритетным вопросом. В работе со взрослыми и старшеклассниками эта модель проектирования используется редко.

Что касается «обратного» проектирования — от целей курса, то оно незаслуженно игнорируется преподавателями, так как вступает в некоторое противоречие с привычкой проектировать на основе контента. Эта

модель предполагает, что сначала четко определяются цели курса и ожидаемые конечные результаты, потом на их основе проектируются критерии оценки их достижения и только после этого подбирается контент и форма учебных активностей [14].

В работах [13, 14] показано, что проектирование от целей является наиболее предпочтительным в большинстве случаев, так как позволяет построить курс оптимальным образом. «Обратное» проектирование дает возможность избежать наполнения курса лишней информацией, которая не способствует быстрому достижению конечного результата, а только увеличивает время изучения, а также позволяет сэкономить другие ресурсы, используя лишь те, которые направлены непосредственно на достижение целей курса.

Учитывая сказанное выше, для проектирования курса «Основы программирования для школьников» была выбрана модель «обратного» проектирования.

4. Применение модели «обратного» проектирования

Конечным результатом обучения по курсу «Основы программирования для школьников» является получение представления о работе программиста, о формировании алгоритмов и их отладке. С учетом требований для курса, которые были описаны во введении, было принято решение в качестве языка программирования использовать JavaScript с графической библиотекой Processing.js [2, 8]. Работа с JavaScript не требует дополнительного ПО, кроме браузера и простого редактора кода (например, Notepad++). Для использования библиотеки Processing.js достаточно скачать файл с ее кодом и разместить его в папке проекта. Все это может быть осуществлено на любом компьютере с практически любыми параметрами и произвольной операционной системой.

Поскольку результатом выполнения заданий станут графические изображения, а также анимированные и интерактивные картинки, то при изучении базовых операторов можно будет избежать скучных задач по выполнению арифметических операций, что послужит дополнительным источником мотивации.

В работе [14] Г. Уиггинсом и Дж. МакТигом (G. Wiggins, J. McTighe) разработаны удобный шаблон проектирования и структурированный алгоритм действий, которые позволяют быстро построить образовательную программу в соответствии с принципами «обратного» проектирования.

В таблице 1 представлены этапы проектирования программы целевого курса на основе шаблона, предложенного Г. Уиггинсом и Дж. МакТигом [14].

Таблица 1

Проектирование курса «Основы программирования для школьников» на основе шаблона, предложенного Г. Уиггинсом и Дж. МакТигом

Шаг 1. Конечные результаты	
Цели курса Каких результатов нужно достичь, к каким изменениям прийти: <ul style="list-style-type: none"> • получение представления о работе программиста, о формировании алгоритмов и их отладке 	Цели на перенос <i>Учащиеся смогут самостоятельно в реальной жизни применить изученное, чтобы:</i> <ul style="list-style-type: none"> • понять, насколько им близка сфера программирования и стоит ли ее выбирать как профессиональную; • применять свои знания для написания небольших алгоритмов по несложным задачам; • углублять свои знания; • изучать другие языки программирования и заранее понимать, что их при этом ожидает (так как базовые понятия и конструкции в различных языках мало отличаются)
Цели на понимание	
Большие идеи <i>После завершения обучения учащиеся будут понимать, что:</i> <ul style="list-style-type: none"> • в программировании очень важен синтаксис языка: каждая точка и каждый символ должны быть на своем месте; • код может выполнять одни и те же действия, но быть грамотно и неграмотно организованным; • достаточно большую часть работы программиста занимают поиск ошибок и отладка 	Ключевые вопросы <i>В ходе обучения учащиеся в той или иной форме ответят на вопросы:</i> <ul style="list-style-type: none"> • Какие рутинные действия составляют основу работы программиста? • Чем отличается грамотно построенный код от неграмотно построенного кода?
Цели на освоение	
Знания <i>Учащиеся будут знать:</i> <ul style="list-style-type: none"> • основные алгоритмические операторы; • основные структуры данных; • принципы построения кода, принципы отладки кода 	Навыки <i>Учащиеся смогут:</i> <ul style="list-style-type: none"> • писать небольшие программы; • правильно выбирать нужные структуры данных; • грамотно организовывать свой код; • применять принципы отладки при поиске ошибок
Шаг 2. Доказательства и оценка	
Критерии оценки <i>Достижение образовательных результатов оценим по критериям:</i> <ul style="list-style-type: none"> • успешное прохождение тестов на знание основных понятий и владение синтаксисом; • соответствие написанных программ техническому заданию; • соответствие проекта чек-листу; • отсутствие синтаксических ошибок 	Доказательства <i>Чтобы показать достижение образовательных результатов, учащиеся будут выполнять такие задачи:</i> <ul style="list-style-type: none"> • писать программы по заданному техническому заданию после просмотра обучающих видео; • искать ошибки в чужих программах; • выполнять тесты <i>Другие доказательства, которые покажут, что образовательные результаты достигнуты:</i> <ul style="list-style-type: none"> • учащийся приходит к четкому пониманию, подходит ли ему эта сфера деятельности, и может объяснить, почему она ему подходит или не подходит
Шаг 3. Учебные активности	
Ключевые учебные активности и стратегии обучения, которые приведут учащихся к успешному прохождению оценивания: <ol style="list-style-type: none"> 1. Для большей вовлеченности обучение будет проходить в виде марафона с начислением очков. 2. В определенное время будет выходить небольшое обучающее видео, в котором разбирается новая тема. Так как результатом выполнения кода в этом курсе будут являться графические изображения или анимация, то в видеоуроках на одной половине экрана должен быть четко виден создаваемый код, а на другой — результат выполнения этого кода. 3. После просмотра видео будет предлагаться задание (так называемый челлендж) по написанию кода. При этом само задание будет содержать либо четкую текстовую постановку, либо показ видео с результатом, который должен получиться (изображение или работа анимации). Результат челленджа в виде html-файла должен быть отправлен преподавателю на проверку. 4. После определенных тем задания должны выдаваться не в виде челленджей, а в виде проектов, где постановка задания допускает проявление творческой мысли, а конечный результат не показан на видео. 5. Несколько раз в течение курса должны происходить встречи в формате видеоконференции (Discord, Zoom) для разбора кейсов с типичными ошибками. 6. Некоторые задания должны быть направлены на поиск ошибок в коде и их устранение. 7. Должно быть организовано тестирование на знание базовых понятий 	

5. Составление программы курса на основе проекта

Таблица 2

Программа курса «Основы программирования для школьников»

Урок/Модуль	Учебные цели	Тезисный план	Домашнее задание
Модуль 1. Начинаем рисовать!			
Урок 1 «Готовим рабочее место и начинаем рисовать»	1. Установить все необходимое ПО. 2. Получить первый видимый результат от рисования в браузере	1. Скачивание Processing.js. 2. Основные блоки HTML. 3. Рисование эллипса	Челлендж 1 «Рисуем снеговика из геометрических фигур»
Урок 2 «Даешь больше геометрических фигур!»	Познакомиться с командами для изображения геометрических фигур	1. Команды для рисования прямоугольников и линий. 2. Команды задания толщины и цвета линий	Челлендж 2 «Снеговик не должен летать» (рисование грунта под снеговиком)
Урок 3 «Красим все, что можно»	Познакомиться с командами заливки	1. Знакомство с палитрой RGB. 2. Команда заливки фона. 3. Команда заливки фигур	Челлендж 3 «Мороз и солнце» (закрашивание цветами черно-белого рисунка со снеговиком, небом и солнцем)
Урок 4 «Еще немного полезных функций»	Познакомиться с дополнительными командами изображения фигур	Команды для изображения точки, треугольника, четырехугольника, дуг и секторов	
Проект 1 «А что у нас на обед?» (изображение еды на тарелке с помощью геометрических фигур)			
Встреча в формате видеоконференции с разбором проекта 1			
Модуль 2. Переменные			
Урок 5 «Введение в переменные»	Познакомиться с понятием переменной	1. Понятие переменной. 2. Понятие значения переменной. 3. Типы переменной	Челлендж 4 «Сделаем медведя страшнее (с помощью использования переменных увеличить глаза и зубы медведя)
Урок 6 «Еще о пользе переменных»	Решение практических задач через переменные	Как выразить все детали рисунка через общие координаты, чтобы при их изменении рисунок перемещался целиком и не деформировался	Челлендж 5 «Помоги лягушке увидеть мир» (дорисовать лягушке рот и глаза так, чтобы при изменении исходных координат лягушка перемещалась вместе с глазами и ртом)
Тест по теме «Переменные»			
Модуль 3. Основы анимации			
Урок 7 «Основы анимирования»	Познакомиться с принципами создания движения	1. Принцип мультипликации. 2. Бесконечный цикл draw()	Челлендж 6 «Восходящее солнце» (построить анимацию с солнцем, которое поднимается из-за горизонта)
Урок 8 «Изменение переменных при анимации»	Научиться увеличивать или уменьшать значения переменных в бесконечном цикле draw()	1. О смысле оператора присвоения значения. 2. Записи вида $x+=3$, $x++$ 3. Задание скорости перемещения	Челлендж 7 «Управляем погодой» (анимирование облаков, закрывающих солнце, так, чтобы они медленно разошлись в разные стороны при одновременном быстром увеличении солнца)
Проект 2 «Падающие звезды» (изображение неба с падающими звездами и небоскребами под ним по техническому заданию)			
Встреча в формате видеоконференции с разбором проекта 2			
Модуль 4. Взаимодействие с мышью			
Урок 9 «Взаимодействие с мышью»	Научиться создавать интерактивные анимированные картинки	Команды управления мышью	Челлендж 8-1 «Сумасшедшая мышь» (мышь должна оставлять след на экране, при этом цвет следа должен меняться).

Окончание табл. 2

Урок/Модуль	Учебные цели	Тезисный план	Домашнее задание
			Челлендж 8-2 «Вкусный помидор» (нарисовать помидор и добиться стирания рисунка мышью)
Модуль 5. Работа с текстом			
Урок 10 «Знакомство с текстом»	Научиться создавать текст на рисунках	1. Команды создания текста. 2. Команды управления цветом текста	
Урок 11 «Анимация текста»	Научиться анимировать текст	Применение уже известных принципов анимирования к тексту	Челлендж 9 «Любимые блюда» (создание отформатированного списка любимых блюд)
Модуль 6. Функции			
Урок 12 «Функции и наследование монстров»	Научиться выделять фрагменты кода для создания функций	1. Обоснование использования функций. 2. Операторы функций	Челлендж 10 «Привет!» (написать функцию, которая выводит приветствие в любой точке холста)
Урок 13 «Передаем параметры в функцию»	Научиться передавать параметры в функцию	1. Формальные и фактические параметры. 2. Разбор примеров	Челлендж 11 «Каждой свинке по луже» (дана функция, рисующая свинку, необходимо разместить несколько свинок в нескольких лужах на холсте)
Урок 14 «Возврат значения функцией»	Научиться создавать функции, которые возвращают значение	Разбор примера с функцией, возвращающей значение	
Урок 15 «Локальные и глобальные переменные»	Научиться понимать разницу между локальными и глобальными переменными	Разбор примера с использованием локальных и глобальных переменных	
Финальный проект			

6. Примеры некоторых учебных активностей

6.1. Пример обучающего видеоурока (модуль 6, урок 14)

На рисунке 2 представлена организация экрана на обучающем видео по теме «Возврат значения функцией». Решается шуточная задача:

«Если известно, что каждый день свинка купается в луже ровно три раза, то сколько раз она купалась, когда ей исполнился один год? два года? три года? четыре года? Надпись с количеством купаний должна появиться под портретом свинки в соответствующем возрасте».

6.2. Пример челленджа по теме «Взаимодействие с мышью» (модуль 4, урок 9, челлендж 8-2)

На рисунке 3 приведен пример челленджа по теме «Взаимодействие с мышью».

6.3. Пример текстовой постановки задачи (модуль 3, проект 2)

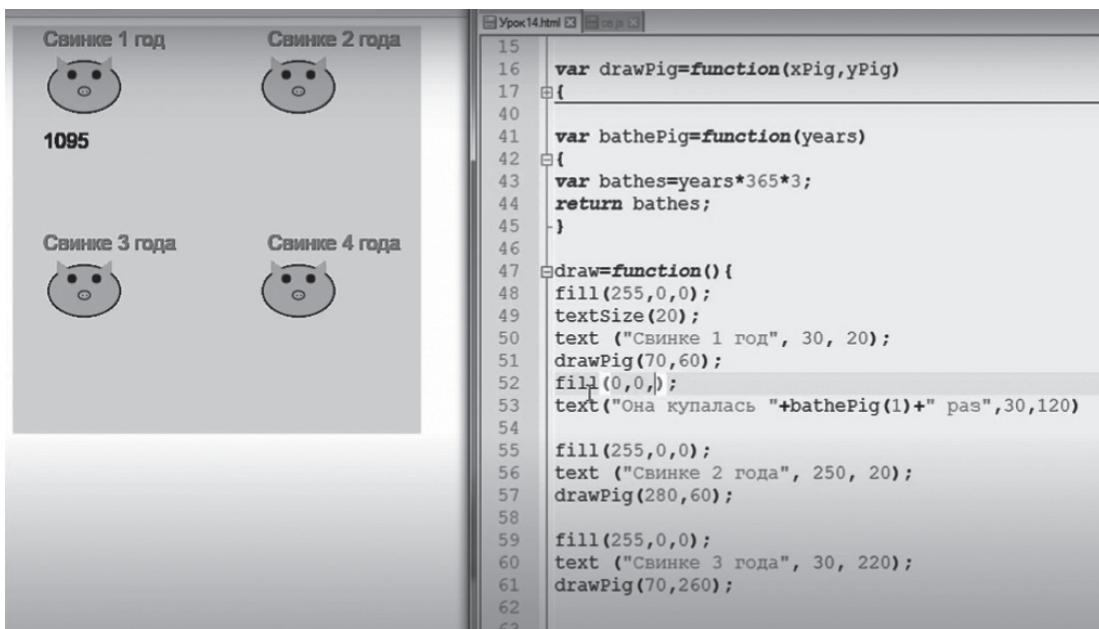
Фрагмент кода, приведенный в листинге, рисует звезду на ночном небе:

```
var xPos = 200;
var yPos = 200;
draw = function() {
  background(29, 40, 115);
  fill(255, 242, 0);
  ellipse(xPos, yPos, 10, 10);
};
```

1. Разместите звезду в том месте холста, откуда она должна начать падение.
2. Выберите, в каком направлении звезда будет падать. Создайте изменение переменных, чтобы анимировать это падение.
3. Создайте другой движущийся в противоположном направлении объект (может быть, облако или другую падающую звезду).
4. Пусть на небе появятся и другие звезды, которые не падают, — у нас будет звездная ночь.
5. Нарисуйте при помощи прямоугольников небоскребы с окнами под этим звездным небом.

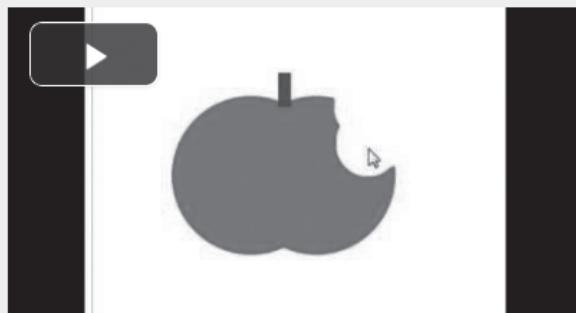
7. Выбор дополнительных технологий для организации курса

Донецкая академия управления и государственной службы достаточно давно использует LMS Moodle для

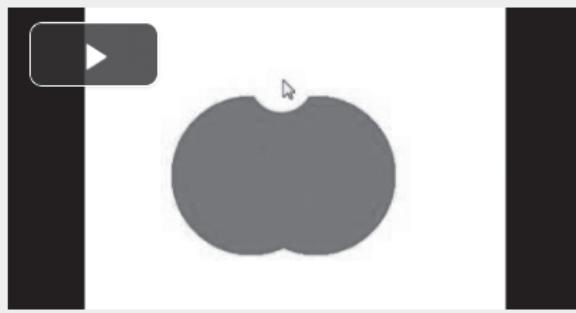


Rис. 2. Пример организации экрана в видеоуроке по теме «Возврат значения функцией»

Нарисуйте помидор и "съешьте" его мышкой, как показано на видео:



Внимание, подвох! Если Вы будете рисовать помидор внутри функции draw(), то при каждом витке цикла помидор будет рисоваться заново. И Вы не можете его "съесть". Как на следующем видео:



Rис. 3. Задание для челленджа в LMS Moodle

сопровождения даже традиционного очного обучения. LMS Moodle является свободно распространяемым ПО с открытым кодом, причем ПО, широко известным в мире и достаточно качественно документированным [3, 5, 12]. Ее установка и использование на собственном сервере являются бесплатными, поэтому выбор в пользу этой LMS был очевиден.

Многие исследователи обращают внимание на то, что современные школьники не мыслят своей жизни без мобильных устройств и социальных сетей, поэтому образовательный процесс может и должен использовать это явление

[1, 4, 6]. В частности, в работах [1, 4] показано, как можно эффективно использовать в процессе обучения чат-боты.

Поэтому еще одной технологией, которая применялась в организации курса, стала **геймификация на основе чат-бота для сети «ВКонтакте»**. Чат-бот был разработан на базе сервиса «Автопилот» [9]. В его задачи входили:

- оповещение учащихся о выходе нового урока;
- напоминание о необходимости выслать домашнее задание на проверку;
- рассылка приглашений в Zoom-конференцию.

8. Выводы

В статье рассмотрен процесс проектирования и запуска программы для онлайн-курса «Основы программирования для школьников» в ЦДП в условиях карантина и сжатых сроков. Обоснованы следующие утверждения:

1. В перечисленных условиях следует использовать верхнеуровневую модель проектирования SAM1 и итеративный подход с пошаговой оценкой и постепенной адаптацией курса под целевую аудиторию.
2. Учитывая идентичность базовых операторов для большинства языков программирования, целесообразно выбирать язык, который не требует сложной и тяжеловесной среды разработки. В то же время желательно, чтобы задания содержали работу с графикой и анимацией, которые стимулируют интерес учащихся. Оптимальным вариантом, удовлетворяющим этим условиям, является JavaScript с графической библиотекой Processing.js.
3. Для проектирования содержания курса оптимальным выбором является применение модели «обратного» проектирования от целей.
4. В качестве дополнительных технологий целесообразно использовать геймификацию на основе чат-бота и LMS Moodle.
5. Курс может быть изучен по ссылке [7]. Параметры авторизации:
 - логин: Gost1411
 - пароль: Conf1411*

Список использованных источников

1. Аристова А. С., Безносюк Ю. С., Ведикер П. К., Воронович Н. Е. Использование чат-ботов в образовательном процессе // Цифровая трансформация общества, экономики, менеджмента и образования. Материалы Международной конференции (Екатеринбург, 05–06 декабря 2019 года). Sedlčany: Ustav personalistiky, 2020. Т. 2. С. 95–99.

2. Архив проекта Processing.js // GitHub. <https://github.com/processing-js/processing-js>

3. Бачанцев И. В., Газейкина А. И., Долгов А. В. Выбор образовательной платформы для создания онлайн курсов по программированию // Актуальные вопросы преподавания математики, информатики и информационных технологий. 2020. № 5. С. 204–211.

4. Гречихин С. С. Дистанционное обучение с помощью образовательных чат-ботов в современных мессенджерах // Балтийский гуманитарный журнал. 2020. Т. 9. № 32 (3). С. 66–68.

5. Документация по LMS Moodle // Официальный сайт Moodle. <http://docs.moodle.org/ru/>

6. Королева Д. О. Всегда онлайн: использование мобильных технологий и социальных сетей современными подростками дома и в школе // Вопросы образования. 2016. № 1. С. 205–222. DOI: 10.17323/1814-9545-2016-1-205-224

7. Курс «Основы программирования (для ЦДП)» // Система дистанционного обучения ГОУ ВПО «ДонАУИГС». <https://elearn.donamp.ru/course/view.php?id=24>

8. Рожков И. В., Федосеева А. П. Особенности обучения программированию на языке JavaScript обучающихся старших классов с углубленным изучением информатики // Наука и образование: проблемы и перспективы — 2017. Сборник материалов региональной научно-практической конференции. СПб.: Амфора, 2017. С. 173–175.

9. Сервис «Автопилот». Официальная страница. <https://skyauto.me/>

10. Allen M. W., Sites R. Leaving ADDIE for SAM: An Agile model for developing the best learning experiences. USA, East Peoria: Versa Press, 2012. 216 с.

11. Brown A. H., Green T. D. The essentials of instructional design. UK, Exeter: Swales & Willis Ltd, 2016. 56 с.

12. Nababan E. B. Implementation of face-to-face online learning system based on audio video, presentation and chat using the moodle e-learning platform // ABDIMAS TALENTA: Jurnal Pengabdian Kepada Masyarakat. 2021. Vol. 6. No. 1. DOI: 10.32734/abdimastalenta.v6i1.5348

13. Richards J. C. Curriculum approaches in language teaching: forward, central, and backward design // RELC Journal. 2013. Vol. 44 (1). P. 5–33.

14. Wiggins G., McTighe J. The understanding by design. Guide to creating high-quality units. Alexandria, Virginia USA: ASCD, 2011.

НОВОСТИ

Платформу «Сфераум» начнут широко использовать в школах с сентября

Замминистра просвещения Российской Федерации Татьяна Васильева сообщила, что информационно-коммуникационная образовательная платформа «Сфераум» с нового учебного года заработает во многих регионах России. Об этом она рассказала на конференции EdCrunch Kuzbass в Кемеровской области.

Татьяна Васильева отметила, что вынужденная работа в дистанционном формате весной 2020 года показала уязвимость системы образования, в том числе из-за отсутствия российской технологии, обеспечивающей связь между учителями и учениками.

«На сегодняшний день «Сфераум» — это полноценно работающая система видео-конференц-связи, система мессенджеров, система управления образовательным процессом, которая, я думаю, с 1 сентября будет широко использована в регионах», — сказала замминистра.

(По материалам, предоставленным пресс-службой Минпросвещения России)

Она подчеркнула, что владение цифровыми компетенциями сегодня актуально для людей всех возрастов. «Развитие компетенций семьи в этом направлении тоже очень важно. Пришло время семьям, у которых недостаточно тех самых базовых компетенций, оказывать помощь, в том числе в школе. Многие в регионах такие программы уже ведут, кто-то в рамках работы с родителями делает такие digital-десанты. Это все преодоление цифрового неравенства в компетенциях», — добавила Татьяна Васильева.

«Сфераум» — это часть цифровой образовательной среды, которая создается Минпросвещения России и Минцифры России в соответствии с постановлением Правительства РФ в целях реализации нацпроекта «Образование». Платформу разработало и запустило совместное предприятие Mail.ru Group и ПАО «Ростелеком» «Цифровое образование». Запуск платформы состоялся 31 марта 2021 года.