

Применение Google Apps Script при реализации балльно-рейтингового оценивания деятельности студентов

Казачкова А.А.

kazachkova.anna@gmail.com

Саратовский государственный университет имени Н.Г. Чернышевского

Аннотация. В статье описывается опыт разработки и применения сценариев на платформе Google Apps Script для реализации балльно-рейтингового оценивания деятельности студентов с использованием Google Таблиц.

Ключевые слова: балльно-рейтинговая система, google таблицы, google apps script.

Балльно-рейтинговая система оценки успеваемости студента – это метод комплексной оценки и управления качеством учебной работы студента, включающий:

1. кумулятивный подход к формированию итоговой оценки по дисциплине;
2. рейтинговые оценки различных видов текущей аудиторной и внеаудиторной (самостоятельной) учебной работы и промежуточной аттестации студентов по дисциплине в семестре.

Основной целью введения балльно-рейтинговой системы был контроль над успеваемостью в течение всего периода обучения, а задачами – формализация процесса оценивания, расширение арсенала видов и форм текущего и промежуточного контроля с сохранением и увеличением прозрачности их применения, а также стимулирование регулярной работы студентов в течение семестра.

Одна из основных сложностей реализации балльно-рейтингового оценивания состоит в том, что с каждым потоком меняется множество важных для подсчёта баллов параметров: количество фактически состоявшихся занятий (которое может отличаться от расчётного по не зависящим от преподавателя и студентов обстоятельствам), количество и состав заданий, сроки представления отчётности и т.д. Даже лекционные, но в особенности семинарские, практические и лабораторные занятия могут включать множество различных типов заданий, отличающихся по количеству, сложности, объёму, критериям оценивания, причём этот состав не остаётся неизменным из года в год. При этом даже небольшое изменение количества заданий определённого типа или соотношения их сложности может привести к тому, что жёстко зафиксированные «расценки» в баллах перестанут работать. Соответственно необходима определённая гибкость: возможность вычислить «стоимость» задания, исходя из общего количества заданий этого типа, возможность настроить «штрафы» за несвоевременную сдачу заданий в зависимости от текущего расписания занятий и т.п.

Одним из удобных способов фиксации и последующего оценивания результатов работы студентов является использование различных электронных таблиц. Электронные таблицы позволяют проводить вычисления с данными, представленными в виде двумерных массивов, имитирующих бумажные

таблицы, они представляют собой удобный инструмент для автоматизации различного рода вычислений. Использование математических формул и других встроенных инструментов зачастую позволяет проводить математическое моделирование в электронной таблице без необходимости применения программирования. Google Таблицы – это облачное веб-приложение для создания и редактирования электронных таблиц. Данное приложение позволяет не только создавать и редактировать таблицы, но также хранить их на сервере и предоставлять доступ с различным уровнем прав как непосредственно к самой таблице, так и к представлению таблицы целиком либо отдельных её листов в формате веб-страниц (без интерфейса Google Таблиц).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	
1		course	Класс	Ia	Ia	Ib	II	II	III	IVa	IVb	IVc	V.Потоки	Отчет	Творч-я	Балл	
2	1	Алгоритмы и структуры данных	C++	3	17	1	20	г	22	ш	п	20	10	1	3	8	33
3	2	Элементы информатики	C#	1	20	12	17	35	к	6	15	13		2		16.4	
4	3	Базы данных	C#	4	14	2	18	25	к	16	7	12		3	7	30	
5	4	Элементы информатики		2	19	8	15	38	п	9	8	12				0	
6	4	Элементы информатики	C++	5	11	3	14	26	п	2	13	8				22	
7	5	Базы данных		12	16	14	16	23	к	22	11	2		3		25	
8	6	Элементы информатики	C++	4	17	13	18	39	п	7	11	19		3		25	
9	7	Элементы информатики		5	16	14	12	36	к	7	9	14				2	
10	8	Элементы информатики	C++	6	12	4	2	28	п	14	9	5				22	
11	9	Элементы информатики	C++	9	12	2	19	33	п	3	17	9		3	4	29	
12		Элементы информатики		10	11	15	10	32	к	4	12	18				2	
13	10	Базы данных	C#	7	14	5	13	30	к	3	15	6		3	8	33	
14	11	Элементы информатики	C#	3	20	18	9	31	п	6	10	15		2		19.5	
15	12	Элементы информатики	C++	8	13	1	11	29	к	11	3	19				6.6	
16		Базы данных	java	7	14	16	14	37	п	1	14	17				5	
17	13	Элементы информатики	C++	10	19	15	8	38	к	5	6	13		3	7	32	
18	14	Элементы информатики	Kotlin	6	15	17	16	30	к	8	16	9				2.5	
19				4	0.5	0.5	2	2	2	1.5	1.5	3	3	2	3	10	35
20															5 + 5		

Рис. 1. Пример таблицы оценивания, в которой используются цвета

В ходе работы с электронной таблицей для оценивания работы студентов может возникнуть желание добавить больше информации в ячейку, например, одновременно указать номер варианта и отметить задание выполненным полностью или частично. В этом случае наглядным решением может стать использование цвета ячейки. Например, более яркий цвет может означать полностью выполненное задание, а более бледный – частичное. Цвет также может использоваться для идентификации преподавателя, который принял задание (если предварительно закрепить за каждым преподавателем определённый цвет), что может быть актуально в случае использования единой таблицы для 2-3 подгрупп одной группы. Дополнительным плюсом использования цвета может быть не только освобождение содержимого ячейки от баллов (что позволит записать в неё номер варианта или какую-то дополнительную информацию), но также возможность не указывать в ячейке количество баллов за задание, если это количество может измениться впоследствии или определяется динамически. Изменение может быть вынужденным, например, если возникла необходимость сократить количество заданий, но может быть наоборот продиктовано желанием более гибко учесть старания студентов, например, дать больше баллов за каждое выполненное задание свыше определённого базового количества.

Цвет фона ячеек в электронных таблицах обычно носит вспомогательный характер: либо используется как декоративный элемент оформления, либо настраивается автоматическое изменение цвета в зависимости от результатов вычислений. Но не наоборот, изменение цвета не влияет на вычисления, в том числе электронные таблицы, как правило, не используют изменение цвета фона ячейки как триггер для пересчёта формул. Однако в вопросах получения, а затем и полноценного использования цвета ячеек может помочь Google Apps Script.

Google Apps Script представляет собой основанный на JavaScript фреймворк для создания веб-приложений на платформе Google Workspace, в частности может использоваться для написания дополнений и сценариев, расширяющих возможности Google Таблиц. Редактор Apps Script можно открыть из меню Extensions в Google Таблицах. Опишем несколько первых шагов, иллюстрирующих процесс составления и редактирования скриптов. Начнём с получения цвета ячейки. Для этого есть подходящий метод `getBackground()`. Чтобы им воспользоваться, нужно создать функцию, например, такую:

```
function getCellBackgroundColor(pRange) {
    return
    SpreadsheetApp.getActive().getRange(pRange).getBackground();
}
```

Листинг 1. Функция, возвращающая цвет фона в hex-формате

Теперь функция может быть использована в ячейке Google Таблицы, но есть несколько трудностей:

1. ячейка-аргумент передаётся в специфическом формате `CELL("address", C16)`
2. значение функции не пересчитывается при изменении фонового цвета ячейки-аргумента, зато пересчитывается при изменении содержимого ячейки – второго аргумента.
3. функция выдаёт цвет в hex-формате, что трудно напрямую использовать в формулах.

Для исправления третьего недостатка можно научить функцию выдавать, к примеру, коэффициент полноты решения, т.е. 1 для тёмного цвета, 0.5 для светлого и 0 для белого. Первый пункт можно исправить, используя выделенные ячейки или же заранее заданные ячейки. Второй пункт самый сложный: изменение фонового цвета ячейки действительно не генерирует какого-либо события, которое можно было бы перехватить и обработать, но и использовать ячейку с фиктивным изменяемым значением тоже не обязательно, можно добавить желаемые действия в меню:

```
function getBackColors() {
    var range = SpreadsheetApp.getSelection().getActiveRange();
    var bgColors = range.getBackgrounds();
    for (var i = 0; i < bgColors.length; i++) {
        range.getCell(i + 1, 1).setValue(bgColors[i]);
    }
}
```

```
}  
  
function onOpen() {  
  SpreadsheetApp.getUi().createMenu('Мои функции')  
    .addItem('Посмотреть выделенные цвета', 'getBackColors')  
    .addItem('Пересчитать баллы', 'calculatePoints')  
    .addToUi();  
}
```

Листинг 2. Добавление пункта меню для определения цветов фона выделенных ячеек

Теперь функция `getBackColors` вызывается с помощью пункта «Посмотреть выделенные цвета» в разделе меню «Мои функции», который (после обновления страницы) появится за последним стандартным разделом `Help`. При вызове она записывает hex-названия соответствующих цветов непосредственно в выделенных ячейках. Эта функция пригодится для составления словаря цветов, который может использоваться в основной функции подсчёта баллов. Алгоритм работы основной функции может быть основан на последовательном построчном обходе ячеек таблицы, определении типа и содержимого ячейки, учёта цвета и накоплении баллов текущего студента.

Созданные на данный момент скрипты не являются универсальными. Для адаптации к новой группе требуется изменить только номер строки с последним студентом, но пока для этого необходимо открыть и исправить код, что несколько затрудняет использование скриптов. Изменения же в заданиях (их количество, состав, добавление новых типов) и принципах подсчёта баллов и штрафов требует более существенных изменений в коде. Также баллы не пересчитываются автоматически при «закрашивании» ячейки, для этого нужно запустить скрипт обновления из меню для всей таблицы, либо выделив необходимые строки. Однако уже в течение нескольких лет скрипты используются автором и коллегами при проведении практических и семинарских занятий по дисциплине «Информационные технологии и программирование» на факультетах КНиИТ и мех-мат, а также практических занятий по теории графов на факультете КНиИТ.