

Информационные технологии в управлении и экономике

2020, № 1

Электронная версия журнала размещена на сайте

<http://it-ugtu.ru>, <http://itue.ru/> и <http://итуэ.рф/>



ISSN 2225-2819

Information technology in management and economics

Информационные технологии в управлении и экономике

2020, № 01 (18), 24.03.2020

Электронная версия журнала размещена на сайте
<http://it-ugtu.ru>, <http://itue.ru/>, <http://итгуэ.рф/>

РЕДАКЦИОННАЯ КОЛЛЕГИЯ

- Рочев К. В., канд. эконом. наук, доцент кафедры вычислительной техники, информационных систем и технологий (ВТИСиТ) УГТУ, главный редактор
- Беляев Д. А., канд. экон. наук, временно исполняющий обязанности ректора УГТУ, президент некоммерческого партнерства «ИТ-Ассоциация Республики Коми»
- Дорогобед А. Н., канд. техн. наук, доцент кафедры вычислительной техники, информационных систем и технологий (ВТИСиТ) УГТУ
- Каюков В. В., доктор экон. наук, профессор кафедры менеджмента УГТУ
- Крестовских Т. С., канд. экон. наук, и. о. директора института экономики, управления и информационных технологий УГТУ, заведующий кафедрой менеджмента УГТУ
- Куделин С. Г., канд. техн. наук, инженер-программист EPAM Systems
- Минцаев М. Ш., доктор техн. наук, проректор по научной работе и инновациям, зав. кафедрой «Автоматизация и транспортная логистика» ГНТУ имени акад. М. Д. Миллионщикова
- Михайлюк О. Н., доктор экон. наук, зав. кафедрой финансов и кредита Уральского государственного горного университета
- Назарова И. Г., доктор эконом. наук, заведующий кафедрой экономики УГТУ
- Павловская А. В., канд. эконом. наук, профессор кафедры менеджмента УГТУ
- Садыкова Р. Ш., доктор экон. наук, профессор, зав. кафедрой экономики и управления предприятием, АГНИ
- Семериков А. В., канд. техн. наук, доцент кафедры вычислительной техники, информационных систем и технологий (ВТИСиТ) УГТУ
- Смирнов Ю. Г., канд. физ.-мат. наук, доцент, зав. кафедрой вычислительной техники, информационных систем и технологий (ВТИСиТ) УГТУ

Журнал выходит 4 раза в год.

Учредитель ФГБОУ ВО «Ухтинский государственный технический университет».

ISSN 2225-2819, свидетельство о регистрации СМИ: Эл № ФС77-65216.

Электронная почта: info@itue.ru

Телефон редакции: +7 (8216) 700-308

Телефон главного редактора: +7 (904) 109-83-18

Статьи, поступающие в редакцию, рецензируются. За достоверность сведений, изложенных в статьях, ответственность несут авторы публикаций. Мнение редакции может не совпадать с мнением авторов материалов. При перепечатке ссылка на журнал обязательна.

Правила для авторов доступны на сайте журнала <http://itue.ru/pravila/>

ОГЛАВЛЕНИЕ

Бильфельд Н. В., Володина Ю. И. Подача произвольного сигнала на вход SIMULINK-модели.....	4
Поторопин В. А., Базарова И. А. АИС «Формирования коллективного расписания для группы обучающихся»	13
Гатин Г. Н. О законе Ципфа	19
Мотова А. С., Рочев К. В. Автоматизированная система виртуального прохождения «эскейп рум» для студии квестов узник	25
Ким М. А., Дорогобед А. Н. Информационная система «Чир спорт Республики Коми»: подсистема участников федерации и показателей их деятельности	31
Сведения об авторах.....	36

**БИЛЬФЕЛЬД Н. В., ВОЛОДИНА Ю. И.
ПОДАЧА ПРОИЗВОЛЬНОГО СИГНАЛА
НА ВХОД SIMULINK-МОДЕЛИ**

УДК 681.5+004.02, ВАК 05.13.00, ГРНТИ 20.00.00

Подача произвольного сигнала на
вход simulink-модели

Non-standard signal input to simulink
model

Н. В. Бильфельд, Ю. И. Володина

N. V. Bilfeld, Yu. I. Volodina

Березниковский филиал Пермского
национального исследовательского
политехнического университета,
г. Березники

Berezniki Branch of Perm National Re-
search Polytechnic University,
Berezniki

В статье рассмотрены способы подачи произвольного (не стандартного) сигнала на вход модели при исследовании поведения систем. Приведено использование блоков библиотеки simulink, а также программная реализация нестандартных сигналов. Рассмотрен пример по использованию пакета по многокритериальному анализу систем управления. Даны рекомендации по сохранению массивов используемых сигналов.

This article is about of methods of non-standard signal inputs to simulink model for investigation of models. Using of simulink blocks is described. Software sources of non-standard signals are given. An example of using of multi-criteria control systems analysis software is presented. Some recommendations for signal arrays saving are given.

Ключевые слова: моделирование, управление, исследование, simulink.

Keywords: modeling, control, re-
search, simulink.

Введение

При моделировании систем управления для исследования динамических свойств объекта на вход системы, как правило, подается один из стандартных возмущающих воздействий [1]. К таким воздействиям принято относить следующие сигналы:

- единичный скачок (функция Хевисайда);
- единичный импульс (бета-импульс);
- гармонический сигнал (как правило, синусоидальный сигнал).

Но при исследовании поведения реальной системы, моделируемой в единицах технологического параметра [2] часто возникает необходимость подать на вход модели не стандартный сигнал. Особенно такая необходимость возникает при проверке точности идентификации при пассивном эксперименте [3]. При таком эксперименте на входе объекта присутствует случайно изменяющийся во

времени сигнал, получаемый с реальных трендов технологических контроллеров. Рассмотрим основные способы моделирования таких систем.

Использование блока *Signal Builder*.

Блок *Signal Builder* находится в стандартной библиотеке *simulink* в разделе *Sources* [4]. Рассмотрим варианты его использования.

Создадим новую модель и поместим в нее указанный блок, как показано на рисунке 1.



Рисунок 1. Блок *Signal Builder*

В командном окне MATLAB введем два вектора, например,

$$X = [0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8]$$

$$Y = [2 \ 5 \ 3 \ 5 \ 8 \ 4 \ 3 \ 5 \ 7]$$

Кликнем по блоку. В результате откроется окно, приведенное на рисунке 2.

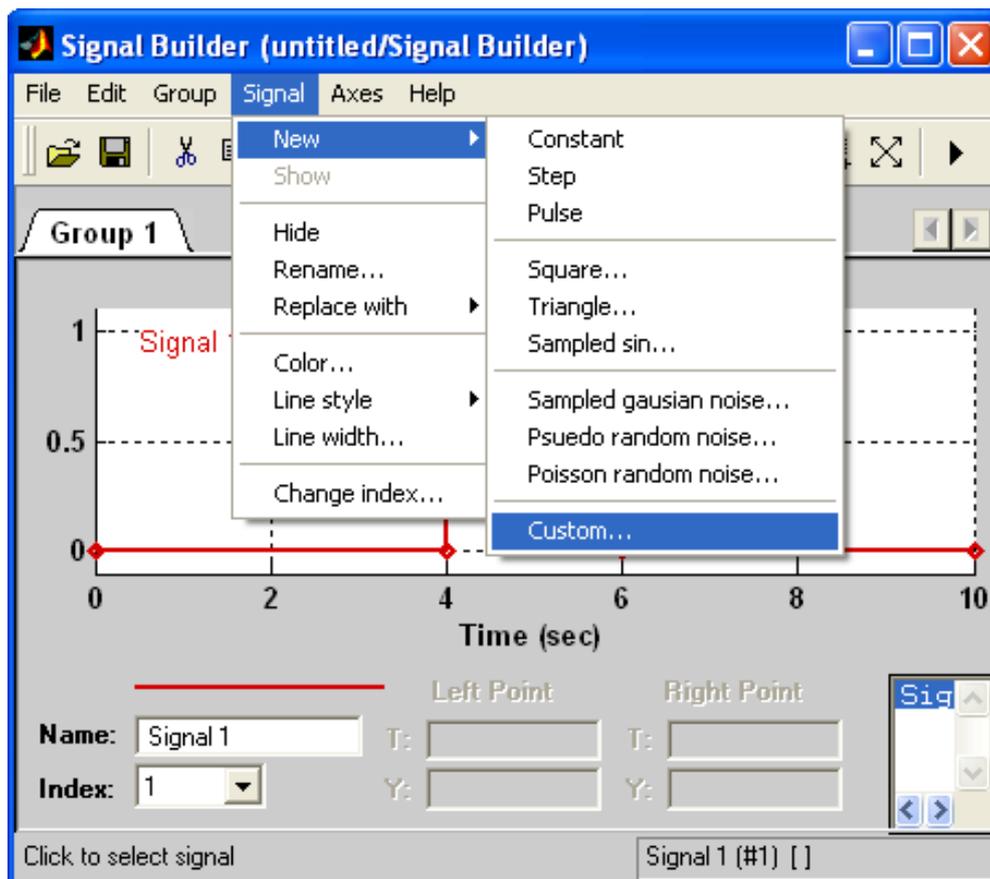


Рисунок 2. Создание сигнала пользователя

Выполним меню «*Signal > New > Custom*». В открывшемся окне введем наши вектора, как показано на рисунке 3.

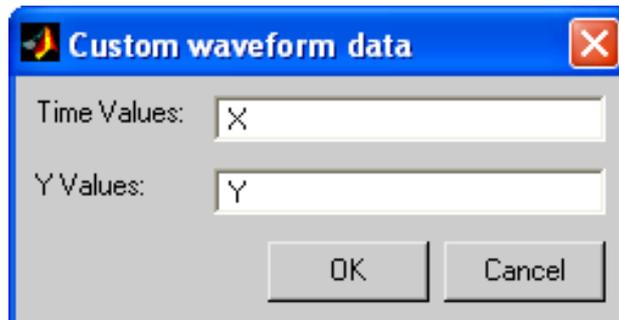


Рисунок 3. Задание векторов

В результате у блока появится второй выход, на который подключим блок Scope, как показано на рисунке 4.

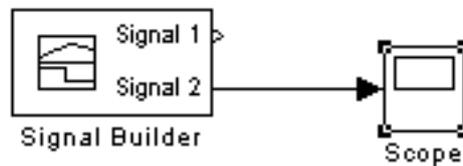


Рисунок 4. Вид блока после задания сигнала

После запуска модели сигнал на осциллографе будет выглядеть, как показано на рисунке 5.

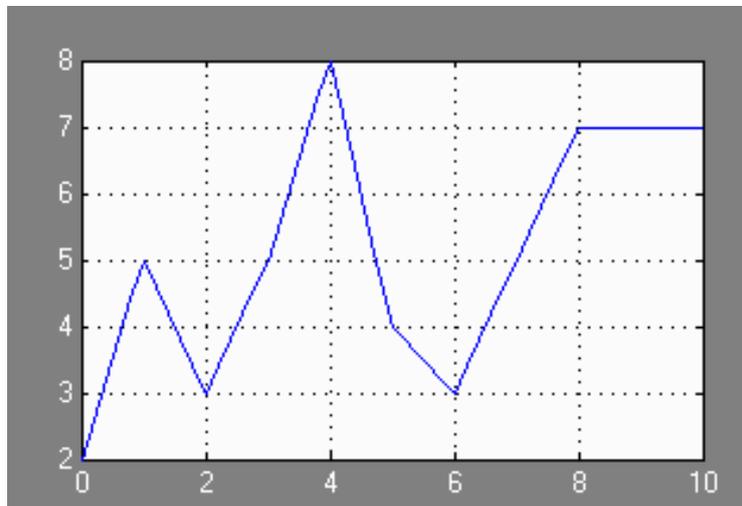


Рисунок 5. Сигнал, заданный векторами. Время моделирования равно 10

С помощью блока *Signal Builder* можно формировать сразу несколько сигналов. В нашем случае на первом выходе блока будет присутствовать единичный скачок (сигнал по умолчанию), а на втором выходе наш сформированный сигнал.

Удалить текущий сигнал можно, выполнив на графике сигнала контекстное меню *Delete*, как показано на рисунке 6. Теперь на выходе блока будет только один сигнал.

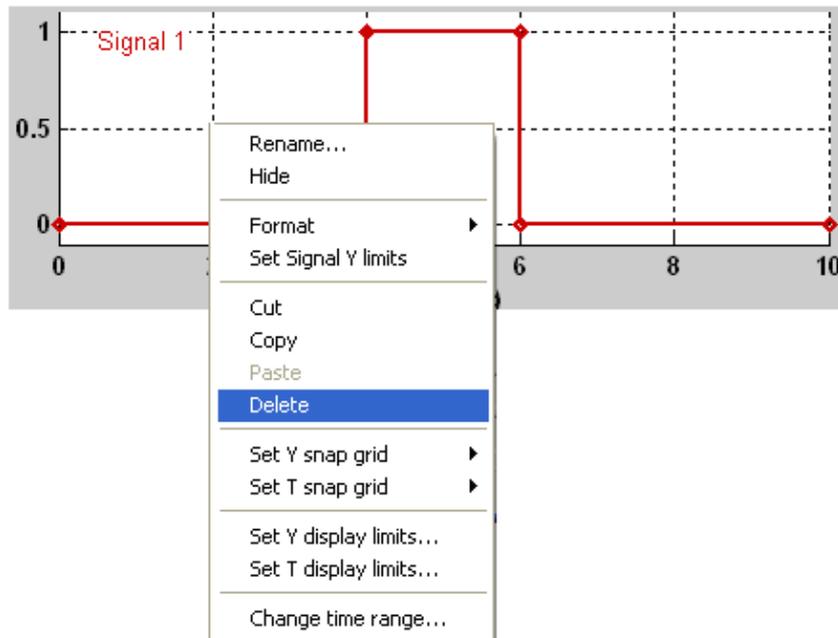


Рисунок 6. Удаление текущего сигнала

Такой сигнал можно подать на вход передаточной функции, как показано на рисунке 7. При этом время моделирования желательно установить равным конечному значению вектора X . На рисунке 5 показан сигнал при времени моделирования, равном 10.

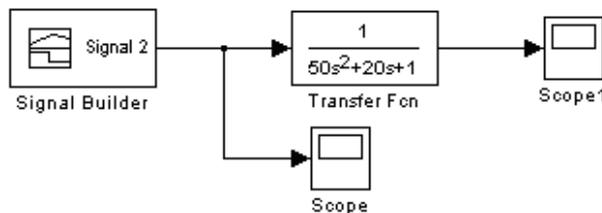
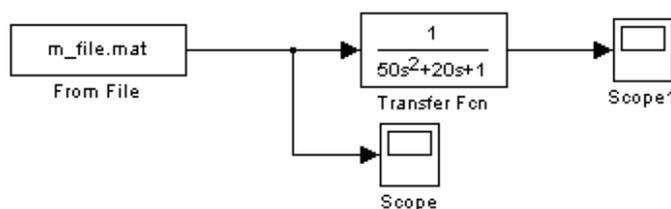


Рисунок 7. Модель с произвольным входным сигналом

Использование блока From file

Блок *From File* также находится в стандартной библиотеке *simulink* в разделе *Sources* и служит для получения сигнала из файла.

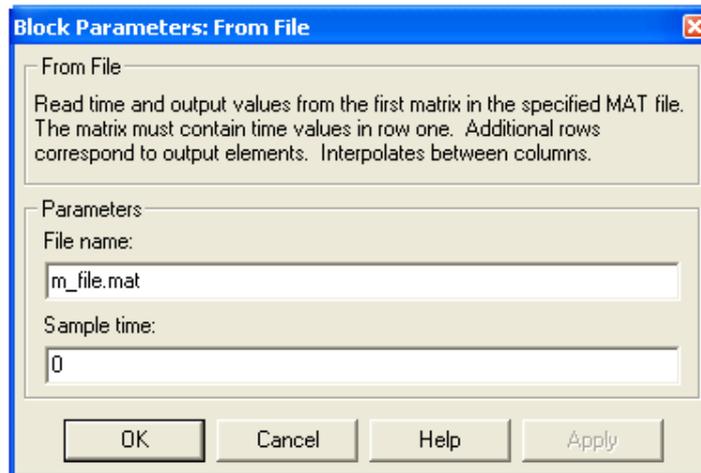
Модель с использованием данного блока приведена на рисунке 8.

Рисунок 8. Модель с блоком *From file* (*Model2*)

Чтобы создать файл, выполним в рабочем окне MATLAB следующие команды:

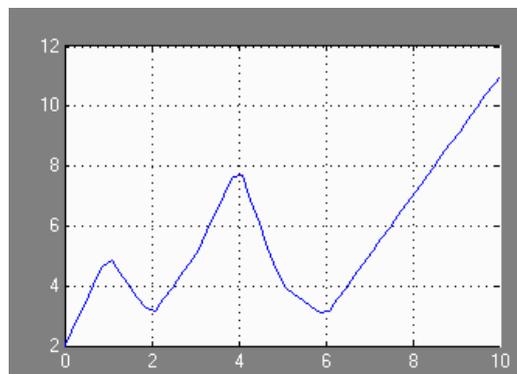
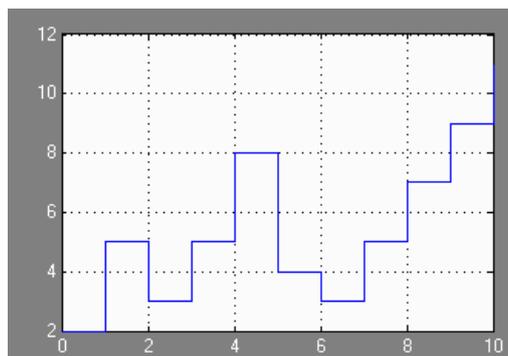
- укажем путь к папке с моделью, например, как $Cd\ d:\backslash asutp\ model;$
- создадим из векторов X, Y двумерный массив, выполнив команду $M = [X; Y];$

- сохраним массив в файле, выполнив команду *Save 'm_file' M*, где *'m_file'* – имя файла. *M* – имя сохраняемого массива;
- кликнем по блоку *from file* и укажем имя файла, как показано на рисунке 9.

Рисунок 9. Настройка блока *from file*

Обратим внимание, что входной сигнал зависит от параметра *Sample time*. Если значение *Sample time* равно нулю, то входной сигнал будет выглядеть, как на рисунке 10. Заметим, что сигнал отличается от сигнала, приведенного на рисунке 5. Углы получаются сглаженными, а сигнал после последней точки продолжает нарастать.

Если значение *Sample time* = 1, то сигнал будет выглядеть, как на рисунке 11.

Рисунок 10. Входной сигнал при *Sample time* = 0Рисунок 11. Входной сигнал при *Sample time* = 1

Использование блока *From workspace*

Блок находится на той же вкладке, что и ранее рассмотренные блоки.

Если уже имеется двумерный массив, то нет смысла сохранять его в файле. Используя блок *From Workspace* можно (рис. 12) указать непосредственно имя массива.

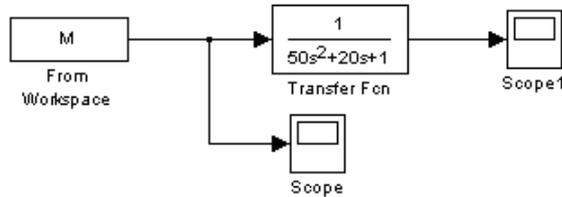


Рисунок 12. Модель с блоком *From Workspace* (*Model3*)

Настройка блока показана на рисунке 13.

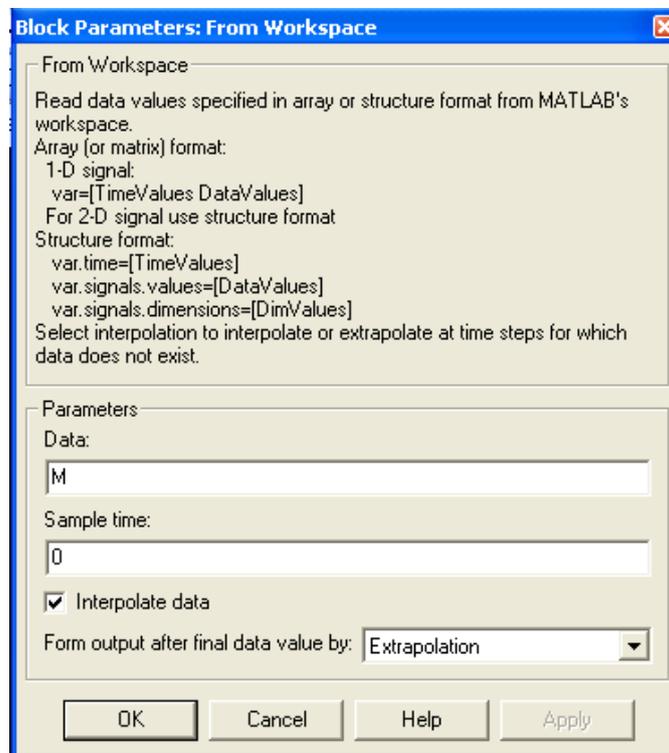


Рисунок 13. Настройка блока *From Workspace*

Если включена опция *Interpolate data* и в раскрывающемся списке выбрана опция *Extrapolation*, то сигналы будут соответствовать сигналам, приведенным на рисунке 10 (при *Sample Time* = 0) и на рисунке 11 (при *Sample Time* = 1).

Значение, выбираемое в раскрывающемся списке «*From output after final data value by*», влияет на «конец» сигнала, если время моделирования больше количества дискретных точек. В частности, если включена опция *Interpolate data* и *Sample Time* = 1, то параметры, выбираемые в раскрывающемся списке, имеют следующий смысл:

- при выборе *Extrapolation* – значение соответствует значению последнего элемента массива;

- при выборе *Setting to Zero* – значение равно нулю;
- при выборе *Holding final value* – значение соответствует значению последнего элемента массива.

Использовавшие команды *Lsim*

Команда *Lsim* позволяет получить сигнал на выходе системы, при изменении входного сигнала по какой-либо известной математической зависимости.

Сначала необходимо отметить, что все команды MATLAB, используемые для построения графиков систем управления (такие как *step*, *impulse*, *nyquist*, *bode* и другие) можно использовать и как функции [5]. Например, если в командном окне MATLAB выполнить строки:

$$W = tf([1], [50 20 1]);$$

$$Step(W), grid$$

то построится график, но если в качестве второй строки указать

$$[Y, X] = step(W)$$

то в результате получатся массивы этого графика (массив ординат указывается на первом месте).

Формат команды *Lsim* имеет вид *Lsim(W, U, T)*, где *W* – передаточная функция системы, на вход которой необходимо подать сигнал; *U* – массив ординат входного сигнала; *T* – массив ординат времени.

Рассмотрим пример, когда на ход объекта необходимо подать перевернутую параболу.

Для формирования параболы выполним следующую последовательность команд:

$$X_1 = linspace(0,100,100);$$

$$A = 0.01; B = 1; Y_1 = -A * X_1^2 + B * X_1;$$

$$X = linspace(0,200,200);$$

$$Y_2 = linspace(0,0,100);$$

$$Y = [Y_1, Y_2];$$

$$plot(X, Y), grid$$

В результате получим график, приведенный на рисунке 14.

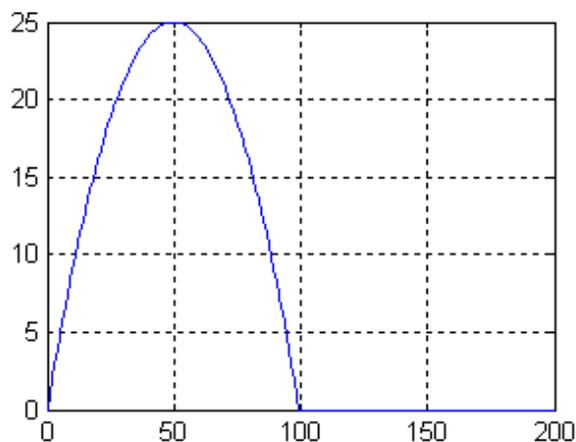


Рисунок 14. Входной сигнал системы

Далее сформируем передаточную функцию и выполним команду *Lsim*:

$$W = tf([1], [50 \ 1]);$$

$$lsim(W, Y, X), grid$$

В результате получим графики, приведенные на рисунке 15.

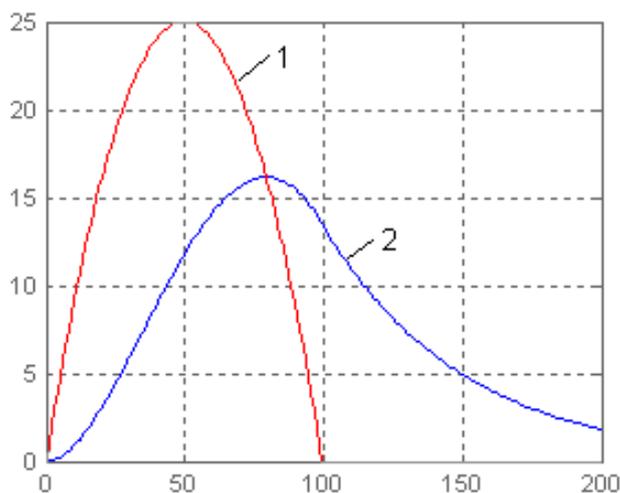


Рисунок 15. Сигналы на входе (1) и выходе (2) системы

Пример использования пакета по многокритериальному анализу систем управления

В заключении рассмотрим формирование случайного сигнала в пакете по многокритериальному исследованию систем управления, разработанного на кафедре АТП БФ ПНИПУ [5]. Данный пакет позволяет работать с различными версиями MATLAB [6].

Для формирования случайного сигнала в пакете необходимо выполнить следующую последовательность действий:

- запустим пакет;
- в поле « $T(K)$ » на форме идентификации укажем необходимый разброс сигнала, например 10;
- выполним меню «Файл» > «Создать» > «Случайный сигнал»;
- кликнем по панели «Дин. Характеристика», чтобы посмотреть на форму сигнала;
- выполнить меню «Файл» > «Передать в MATLAB» > «Оба столбца» и введем имя массива, например *SysXY*;
- перейдем на форму «Работа с моделями *simulink*»;
- выполним меню «Модель» > «Открыть» и откроем модель «*Zaton \ Model3*»;
- запустим модель;
- на форме в поле *Scope* введем *ScopeData2*;
- нажмем кнопку «График».

В результате получим график выходного сигнала, приведенный на рисунке 16



Рисунок 16. График выходного сигнала модели, при случайном сигнале на входе

Из рисунка видно, что чем больше инерционность передаточной функции, тем меньше колебаний в выходном сигнале.

Сохранение рабочей области и переменных в файлах

И последнее, на что стоит обратить внимание, это как можно сохранить созданные сигналы для последующей работы. Для этого можно использовать две возможности MATLAB – это сохранение рабочей области и сохранение переменных.

Сохранение рабочей области подразумевает автоматическое сохранение всех переменных, используемых в данном сеансе работы. Сохранение рабочей области осуществляется из окна рабочей области (*Workspace*) с помощью кнопки «*Save Workspace As*». Восстановление рабочей области осуществляется по кнопке «*Load Data File*».

Сохранение переменных подразумевает сохранение в файле конкретной переменной (вектора, массива и т.д.).

Сохранения переменной, осуществляется командой *Save 'm_file' M*, где *'m_file'* – имя файла, *M* – имя сохраняемой переменной. В результате переменная сохраняется в формате **.mat*. Чтение переменной из файла осуществляется командой *Load M*, где *M* – имя переменной.

Выводы

Подача на вход *Simulink* моделей различных сигналов позволяет осуществлять разностороннее тестирование моделей на предмет выявления различных ошибок и их устойчивой работы.

Описанные выше подходы многократно использовались при исследовании систем управления на практике. Разработанный на кафедре пакет позволяет применять удобный интерфейс, для управления *Simulink* моделями различных версий MATLAB [6]. Так в частности была разработана и протестирована достаточно сложная модель процесса грануляции, насчитывающая более 700 блоков библиотеки *Simulink* [7]. Также, различные сигналы подавались на вход модели сепарации губчатого титана, для выявления ее правильной работы [8].

Список использованных источников и литературы

1. Гайдук Ф. Р., Беляев В. Е., Пьявченко Т. А. Теория автоматического управления в примерах и задачах с решениями в MATLAB. – М.: Лань, 2011. – 464 с.
2. Бильфельд Н. В. Моделирование в реальном режиме времени в среде MATLAB // *Juvenis scientia*. – 2016. № 2. – С. 16-18.
3. Бильфельд Н. В. Пассивная идентификация объектов управления средствами TOOLBOX IDENT // *Juvenis scientia*. – 2016. № 3. – С. 4-7.
4. Дьяконов В. П. MATLAB6/6/1/6/5+SIMULINK 4/5. – М.: Солон-Пресс, 2002. – 768 с.
5. Бильфельд Н. В., Многокритериальное исследование систем управления. – Пермь : ПНИПУ, 2015. – 436 с.
6. Бильфельд Н. В., Митюков Е. А. Одновременная работа с различными версиями пакетов MATLAB как серверов OLE-автоматизации // *Новый университет. Серия: Технические науки*. – 2015. № 9-10 (43-44). – С. 67-70.
7. Затонский А. В. Программные средства глобальной оптимизации систем автоматического регулирования – М: ИЦ РИОР, 2013. – 136 с.
8. Затонский А. В., Бильфельд Н. В. Применение самоорганизующихся систем при управлении сложными процессами // *Проблемы теории и практики управления*. – 2007. №12. – С.70-74.

List of references

1. Gaiduk F. R., Belyaev V. E., Pjavchenko T. A. Theory of automatic control in examples and problems with solutions in MATLAB. - M.: Lan, 2011. – 464 p.
2. Bilfeld N. V. Modeling in real time in the environment of MATLAB // *Juvenis scientia*. – 2016. № 2. – P. 16-18.
3. Bilfeld N. V. Passive identification of asset management tools TOOLBOX IDENT // *Juvenis scientia*. – 2016. №3. – P. 4-7.
4. Dyakonov V. P. MATLAB6/6/1/6/5+SIMULINK 4/5. – M.: Solon-Press, 2002. – 768 p.
5. Bilfeld N. V., Mnogokriterial'noye issledovaniye sistem upravle-niya. – Perm: PNIPU, 2015. – 436 p.
6. Bilfeld N. V., Mityukov Ye. A. Odnovremennaya rabota s razlichny-mi versiyami paketov MATLAB kak serverov OLE-avtomatizatsii // *Novyy universitet. Seriya: Tekhnicheskiye nauki*. – 2015. № 9-10 (43-44). – P. 67-70.
7. Zatonsky A. V. Software for global optimization of automatic control systems – M: IC RIOR, 2013. – 136 p.
8. Zatonsky A. V., Bilfeld N. V. The use of self-organizing systems in the management of complex processes // *Problems of control theory and practice*, – 2007. №12. – P. 70-74.

ПОТОРОПИН В. А., БАЗАРОВА И. А. АИС «ФОРМИРОВАНИЯ КОЛЛЕКТИВНОГО РАСПИСАНИЯ ДЛЯ ГРУППЫ ОБУЧАЮЩИХСЯ»

УДК 004.91:37, ВАК 05.13.01, ГРНТИ 50.49.37

АИС «Формирования коллективного расписания для группы обучающихся»

В. А. Поторопин, И. А. Базарова

Ухтинский государственный
технический университет, г. Ухта

В статье рассматривается разработка автоматизированной системы «Формирования коллективного расписания для группы обучающихся». В ходе разработки основными приоритетами были: агрегация расписания, оперативное уведомление студентов. В результате было разработано готовое к внедрению асинхронное серверное приложение.

Ключевые слова: Расписание, мессенджеры, уведомление.

AIS «Organization collective schedule for a group of students»

V. A. Potoropin, I. A. Bazarova

Ukhta state technical university,
Ukhta

The article examines the development of an automated system «Organization collective schedule for a group of students». During development main priorities were: schedule aggregation, quick student notification. As a result, an asynchronous server-ready application was developed.

Keywords: Schedule, messenger, notification.

Введение

Основным источником информации о расписании в Ухтинском государственном техническом университете является официальный сайт университета. За несколько недель до начала нового семестра диспетчер составляет первую версию расписания, затем, кафедры, кооперируясь друг с другом посредством телефонных звонков, вносят свои исправления. Исправленные расписания отправляются обратно диспетчеру, который проверяет их, утверждает и выкладывает на официальном сайте университета.

Тем не менее, в силу одних и тех же обстоятельств (болезней преподавателей, неудобного времени пар, увольнений) расписание на сайте быстро теряет актуальность. В скором времени единственным достоверным источником расписания становятся бумажные версии расписаний, вывешенные на каждой из кафедр.

Кроме расписания занятий, часто возникает необходимость прислать группе учебный материал, сообщить о заменах в расписании или о любой другой важной информации. В этих случаях используются электронные ящики старост каждой из групп или дополнительные записи на информационных стендах кафедры. Для подобных, ежедневно возникающих задач, необходимы удобные и надежные каналы передачи информации, которыми не могут считаться бумажные информационные стенды.

Описанные выше процессы не автоматизированы, а большинство известных методик автоматизации расписания подходят не полностью из-за географической разнесенности корпусов УГТУ. В связи с этим, разработка системы является актуальной для университета.

Предпроектное обследование

В результате изучения предметной области были выявлены существенные проблемы, с целью решения которых было решено создать систему, автоматизирующую оповещение студентов и предоставление актуальной версии расписания в реальном времени.

При обзоре аналогов системы руководящим мотивом было сравнение постановок задач, лежащих в основе, разрабатываемой и оцениваемых в качестве аналогов систем и выявление потенциально полезных функций, которые не вошли в постановку задачи на разработку системы.

– **stud.rucoop.ru**: общее расписание для всех филиалов университета, объединенное на одном сайте. Для доступа не требуется регистрация, но каждый раз необходимо вводить все параметры заново, так как они не сохраняются в адресной строке.

– **it-institut.ru**: объединенная онлайн-платформа для формирования и отображения расписания [1]. Имеет множество подключенных вузов по всей России. Расписание формируется техником каждого вуза, доступ осуществляется без регистрации, посредством ввода параметров для своей группы.

Оба примера являются типовым решением для большинства университетов. В таком формате для расписания необходим техник, поддерживающий его в актуальном состоянии, в противном случае неизбежна его деградация, которая приводит к возникновению спорных ситуаций, потери времени, необходимости уведомлять каждую группу отдельно о заменах и проводимых в университете событиях.

Более подробный обзор и сопоставление аналогов представлены в таблице 1.

Проектирование информационной системы

В ходе анализа предметной области была построена диаграмма потоков данных «Как будет» (рис. 1), отражающая состояние процесса, подвергнувшегося автоматизации.

Для реализации требований, определенных в предпроектном обследовании, была выбрана модель чат-бота. Чат-бот – компьютерная программа, имитирующая речевое поведение человека при общении с одним или несколькими собеседниками. Чат-бот ведет коммуникацию от лица компании или бренда с целью упростить онлайн-общение (предоставить актуальную информацию в наиболее оперативные сроки) [2].

Таблица 1. Сравнение аналогов

	АИС «Формирования коллективного расписания для группы обучающихся».	Расписание УГТУ	stud.rucoor	it-institut	studify
Абонентская плата	Нет	Нет	Нет	Да	Да
Мобильное приложение	Нет	Нет	Нет	Нет	Да
Использование социальных сетей	Да	Нет	Нет	Нет	Нет
Оповещение студентов и преподавателей	Да	Нет	Нет	Нет	Да
Возможность изменения расписания	Техники, Преподаватели, Диспетчер	Диспетчер	Диспетчер	Диспетчер	Диспетчер
Проверка на ошибки в расписании	Да	Нет	Нет	Да	Нет
Текущее расписание преподавателя	Да	Нет	Нет	Да	Да
Расписание консультаций и приема задолженностей	Да	Нет	Нет	Нет	Да

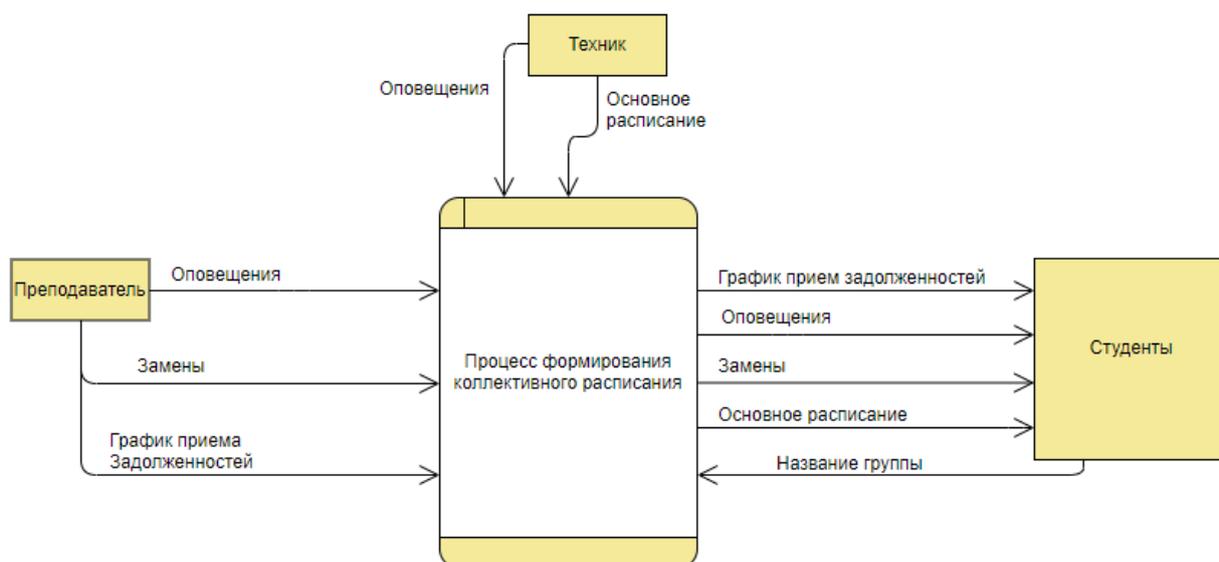


Рисунок 1. Контекстная диаграмма «Как будет»

Результаты разработки системы

Результатом разработки системы является чат-бот, написанный на языке python.

Приложение имеет клиент-серверную архитектуру. На данный момент боты – это в первую очередь возможность сверхоперативно доставлять контент напрямую потребителю [3]. Быстрая скорость ответа была одним из главных факторов, учитывающихся при разработке бота. Благодаря тому, что во время проектиро-

вания не было обнаружено частей, требующих значительных затрат процессорного времени, была выбрана асинхронная структура приложения, которая позволяет значительно ускорить программы, привязанные к системам ввода\вывода.

Пример работы приложения представлен на рисунке 2.

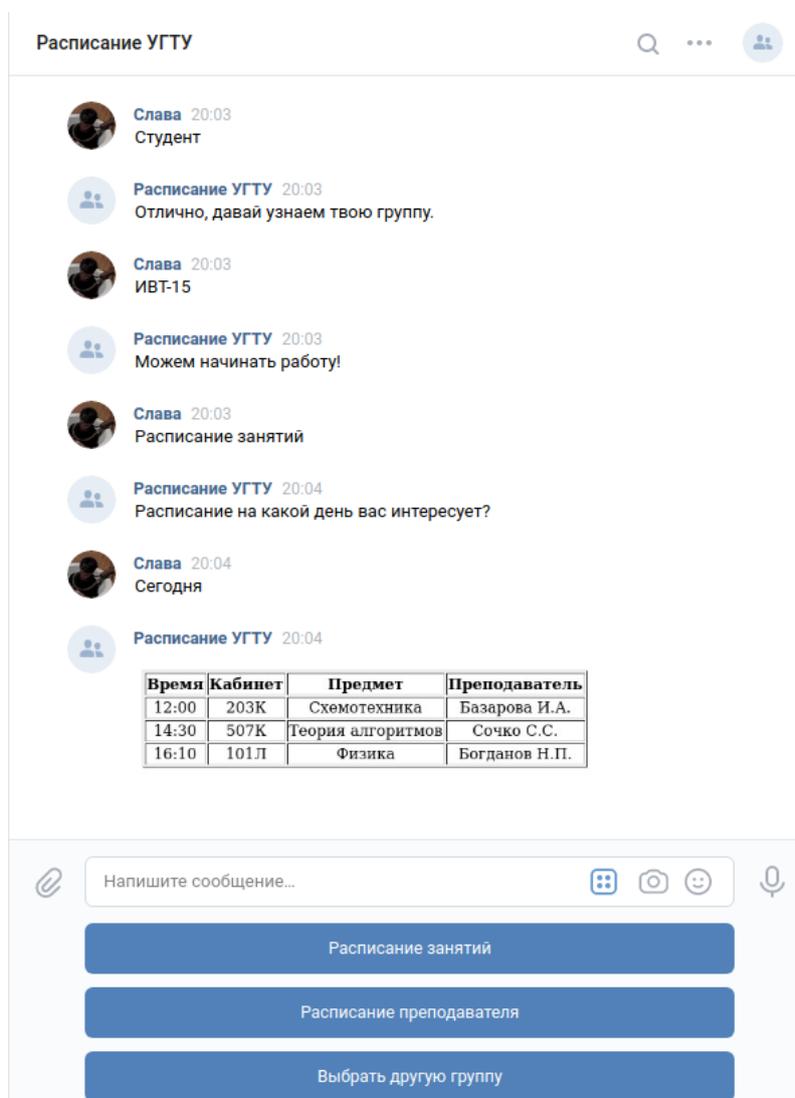


Рисунок 2. Пример работы приложения

Заключение

На основе анализа предметной области были сформированы основные цели и требования к созданию системы. Также было написано серверное приложение, которое выполняет поставленные задачи. Полученное приложение позволит студентам УГТУ всегда иметь под рукой информацию об актуальном расписании, кроме того, преподаватели и технические работники получат возможность оперативно сообщать студентам важную информацию. В будущем систему можно расширить, добавив новый функционал или улучшив текущие возможности.

Список использованных источников и литературы

1. Электронное расписание [Электронный ресурс]. – Режим доступа: <http://www.it-institut.ru> (Дата обращения: 05.05.2019)
2. Википедия [Электронный ресурс]. – Режим доступа: <https://www.wikipedia.org> (Дата обращения: 10.05.2019)
3. Иванов А. Д. Чат-бот в telegram и вконтакте как новый канал распространения новостей // Вестник волжского университета им. В. Н. Татищева. 2016.

List of references

1. Electronic schedule, <http://www.it-institut.ru>, accessed May 05, 2019.
2. Wikipedia, <https://www.wikipedia.org>, accessed May 10, 2019,
3. Ivanov, A. D., “Chatbot in telegram and vkontakte as the new channel of news distribution”, Bulletin of the Volga University named after V. N. Tatishchev, 2016.

ГАТИН Г. Н.
О ЗАКОНЕ ЦИПФА

УДК 81.32, ВАК 05.13.01, ГРНТИ 20.00.00

О законе Ципфа

About Zipf's Law

Г. Н. Гатин

G. N. Gatin

Ухтинский государственный
технический университет, г. Ухта,

Ukhta state technical university,
Ukhta

Автор производит проверку литературных, программных и случайных текстов на соответствие закону Ципфа. Показывает, что существуют тексты, не отвечающие этому закону. Обращает внимание, что использование в Internet Закона Ципфа для проверки естественности текста бесполезно.

The author checks literary, programmatic and random texts for compliance with Zipf's law. Shows that there are texts that do not meet this law. It is noteworthy that the use of the Zipf Law on the Internet to verify the naturalness of the text is useless.

Ключевые слова: закон Ципфа, распределение, текст, язык, частота, слово.

Keywords: Zipf's law, text, distribution, language, frequency, word.

Введение

Обычно, известие о том, что из всего словарного запаса языка мы используем где-то ~2000 слов, первый раз звучит достаточно удивительно. Выполняя краткий анализ этого утверждения, а именно, удалив из этих ~2000 слов предлоги, союзы, частицы или одно-двухбуквенные (а иногда более) слова, мы понимаем, что наш словарный запас грозит сравняться со словарём знаменитой Элочки - людоедки – это уже шокирует.

Тогда-то мы и ищем обоснование всего этого. Оказывается это закон Ципфа.

Например, в книге В. А. Лапшина "Лекции по математической лингвистике" [1]. В. А. Лапшин приводит его в форме:

$$P_n \approx \frac{1}{n^a} \quad (1)$$

В. А. Лапшин приводит этот закон для такого объекта как "мешок слов русского языка". Другими словами, закону Ципфа подчиняется весь язык, весь словарный запас языка. Но в печатной литературе вы не найдёте обоснования или доказательства закона Ципфа – это просто эмпирическая закономерность.

Теоретический анализ

Как профессиональный программист я привык все утверждения литературы по программированию не принимать на веру, а проверять. Не факт, что скопированная программа или методика "пойдут" на вашей машине или в вашем окружении.

Формулировка же закона Ципфа предельно проста, что подталкивает вас к решению проверить закон. К тому же, до 2005 года (см. ниже) это была недоказанная наблюдаемая статистическая закономерность.

Но, конечно, прежде всего выполняем обзор.

Открыл закон, оказывается, французский стенографист Жан Батист Эсту (фр. Jean-Baptiste Estoup) в 1908 году, что говорит в пользу применения закона ко всему словарному запасу языка. Википедия [2] приводит следующий график частот слов:

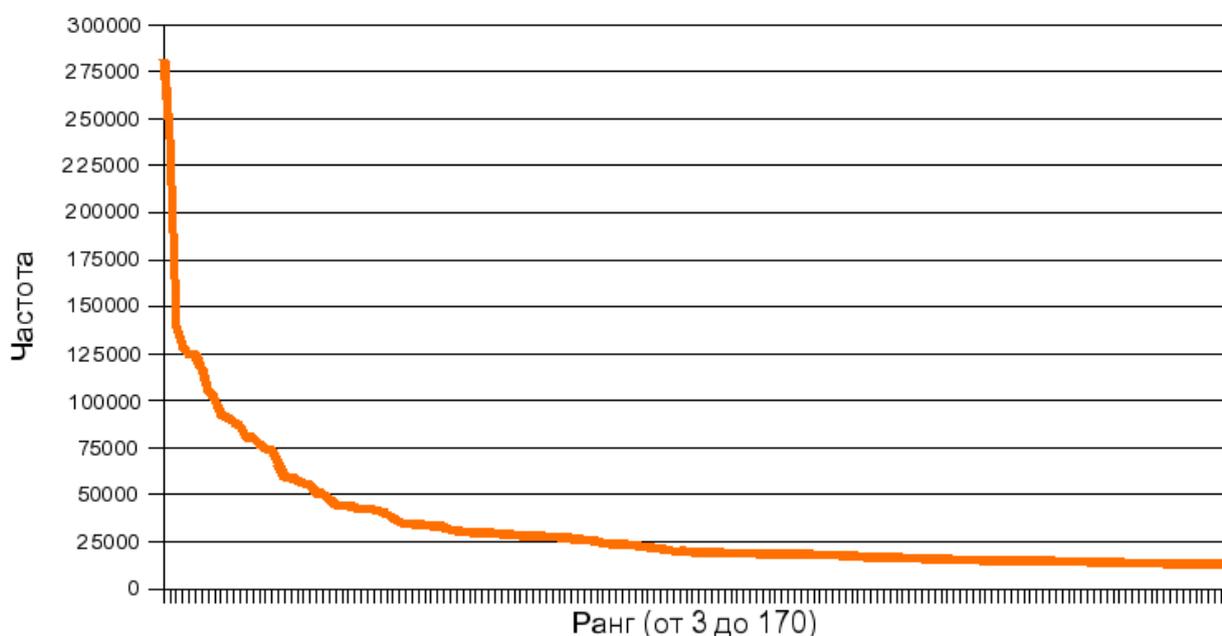


Рисунок 1. Закон Ципфа. График для для частот слов из статей русской Википедии с рангами от 3 до 170. [200]

Форма закона здесь немного иная, но, впрочем, та же:

$$P_i = P_1 / i \quad (2)$$

Интересно, что этому закону подчиняется население городов: "город с самым большим населением в любой стране в два раза больше, чем следующий по размеру город, и так далее" [2]; доходы населения, ну и, конечно же, тексты.

Для городов в 1999 году экономист Ксавье Габэ показал, что если города будут расти случайным образом с одинаковым среднеквадратичным отклонением, то в пределе распределение будет сходиться к закону Ципфа. [2]

Но, самое главное, "Объяснение закона Ципфа, основанное на корреляционных свойствах аддитивных марковских цепей (со ступенчатой функцией памяти) было дано в 2005 году" [2].

Это утверждение, выпускает дух из вашего желания как-то проверить закон Ципфа. Но дальнейшая работа в Internete снова вас вдохновляет.

Оказывается, что часть сайтов проверяет тексты на естественность, считая, что чем более текст соответствует закону Ципфа, тем он естественней. С другой стороны, существует статья американского специалиста по биоинформатике [3], в которой он утверждает, что закону Ципфа подчиняются чисто случайные тексты. И если это утверждение верно, то вышеуказанные проверки бесполезны.

Безусловно, проверить живую речь на закон Ципфа достаточно трудоёмкое занятие, хотя и интересное. Ну, например, сколько слов мы говорим в сутки? В сутках 1620 мин. Исключаем 8 часов сна и один час на гигиену, остаётся 1080 мин. Если мы произносим, хотя бы десять слов в минуту, то за сутки обеспечено 10 800 слов, что позволяет выполнять проверку закона уже после первых суток. Я бы снизил число слов в четыре раза, что даёт в общей сложности время набора слов примерно от четырёх до пяти - семи дней. Основная проблема набрать статистически значимую выборку. Принципиально всё это считается: для массива из N слов ($\approx 250\ 000$ слов русского языка), выборка объёмом n слов с вероятностью P даст отклонение не более k слов.

В общем-то, мы и так знаем, что в обыденной своей речи используем где-то 2000 слов, но подчиняется ли эта выборка закону Ципфа? Каких-либо данных на ту тему я не обнаружил. С другой стороны, кажется, что если бы мы чаще всего использовали некое значимое достаточно длинное слово, то это слово мы бы знали. Но таких слов никто не вспомнит!

Интуитивно понятно, что наиболее часто мы используем вспомогательные одно- и двухбуквенные слова (ну если быть педантичным, это требует проверки).

Другими словами, если вы собираетесь заниматься модной сегодня обработкой текста, то у вас должны быть все эти данные. Это веский аргумент в пользу проверки закона Ципфа.

Кажется, чего автор кипятится? Возьмите частотный словарь русского языка. Какие проблемы? А проблем-то достаточно. Вы не найдёте общедоступного словаря русского языка, который можно подключить к вашей программе и использовать его как источник слов, не данных даже, а просто слов. Автор имеет опыт обработки словаря из Internet. Некоторые слова набраны латиницей, ошибки в написании, пропущенные пробелы и т.п.

Вот почему мы проанализировали некоторое количество текстов.

Опять-таки, прежде чем заниматься, некоторой рутинной работой, необходимо обдумать: «А что, собственно, мы собираемся получить?»

Начнём с вопроса: Существуют ли тексты, не подчиняющиеся закону Ципфа? Закон сформулирован для языка, а значит, таких текстов не существует.

Тем не менее, такие тексты есть! В любом орфографическом словаре распределение слов равномерное. С толковыми и двуязычными словарями сложнее. Распределение слов, скорее всего, будет стремиться в закону Ципфа, но как велико отклонение?

Интуитивно ясно, что закону Ципфа не подчиняются: Простые словари, Каталоги, Сметы, Нормативы.

Со случайными текстами сложнее: всё сильно зависит от алгоритма генерации текста, результаты исследований представлены ниже в таблицах.

Экспериментальная часть

Для проверки были выбраны: две статьи автора, как короткие тексты, несколько литературных произведений, библия и толковый словарь русского языка (таблица 1). Кроме того, были добавлены авторские исходные модули на языке "C++", как смесь искусственного языка и имён, задаваемых программистом.

Таблица 1. Коэффициенты регрессии для выбранных текстов

	Всего слов	b_1	b_0	угол наклона	R^2	F
Статья автора "К критике компетенций"	2617	-0.58376	3.9258	120.275	0.890809	8.15825
Статья автора "Философия языка 'С' "	4117	-0.63439	4.4943	122.391	0.902867	9.2951
Сборник рассказов Габриэль Гарсия Маркес	81930	-0.8110	7.6787	129.043	0.93384	14.1162
Русские народные сказки	117170	-1.11948	10.593	138.226	0.949559	18.8253
М. А. Булгаков Мастер и Маргарита	118154	-0.85071	8.2974	130.388	0.948383	18.3734
Сказки 1000 и одна ночь	248974	-1.07892	10.653	137.174	0.974394	38.0526
Библия	714243	-1.1572	12.292	139.168	0,978777	46.1183
Толковый словарь русского языка (Ожегов С.И. Шведова Н.Ю.)	1140220	-0.9206	10.651	132.633	0.96212	25.403
Программы на языке C++	9434	-1.1225	7.9024	138.305	0.97085	33.3034

В таблице представлены прямые регрессии, подбираемые методом наименьших квадратов, где:

- ось ординат – логарифмы частот слов;
- ось абсцисс – ранги (номеров слов, отсортированных по убыванию) – n ,

уравнение подбиралось в виде:

$$chastota = b_1 * n + b_0;$$

➤ R^2 – (квадрат) множественный коэффициент корреляции показывает процент отклонения от среднего (\bar{Y} -ср) и объясняется уравнением регрессии;

➤ при анализе - критерия Фишера F выявлено, что $F > 1$, поскольку число степеней свободы чрезвычайно велико, то является достаточным.

Написанное автором ПО, считывало текст, подсчитывало число слов и вычисляло частоту встречаемости слов. Затем методом наименьших квадратов определялись коэффициенты b_1 и b_0 регрессионной прямой зависимости логарифма частоты от ранга слова. Адекватность формулы проверялась по множественному коэффициенту корреляции (R^2) [3] и критерию Фишера (F). Результаты приведены в таблице 1. Результаты однозначно подтверждают истинность закона Ципфа для литературных текстов и алгоритмического языка "C++".

Была выполнена проверка на закон Ципфа случайных текстов. Однако, что считать "чисто случайным текстом"? Можно генерировать "случайный" текст, так чтобы сохранять частоту встречаемости букв русского языка, а слова случайные последовательности этих букв. Можно выбирать случайным образом слова из словаря русского языка (в этом случае, скорее всего, гипотеза о соответствии текста закону Ципфа не подтвердится, поскольку все слова будут распределены равномерно).

Автор ориентировался на "чисто случайный текст". Сначала генерировалась длина слова, затем буквы, входящие в слово; буквы и длины слов распределены равномерно. Результаты приведены в таблицах 2 – 5. Видим, что случайный текст закону Ципфа не соответствует. Таким образом, сообщение Вэньтянь Ли не подтверждается, с оговоркой – "на нашей модели случайного текста".

Таблица 2. Коэффициенты регрессии для случайных текстов

Всего найдено слов	Коэффициенты прямой регрессии	Угол наклона	Степеней свободы для критерия Фишера	Коэффициент корреляции	Критерий Фишера
40000	$b_1 = -0.0646027$ $b_0 = 0.632828$	93.6963	39998	$R^2 = 0.20381$	$F = 0.255981$

Таблица 3. Частоты встречаемости слов в зависимости от длины

длина слова	1	2	3	4	5	6	7	8	9	10
число слов	1382	1404	1445	1353	1414	1397	1398	1379	1407	1432
длина слова	11	12	13	14	15	16	17	18	19	20
число слов	1428	1373	1460	1433	1353	1358	1359	1475	1401	1389
длина слова	21	22	23	24	25	26	27	28	29	
число слов	1398	1442	1357	1370	1410	1406	1428	1423	1344	

Таблица 4. Коэффициенты прямой регрессии для случайных текстов

Коэффициенты прямой регрессии	Угол наклона	Степеней свободы для критерия Фишера	Коэффициент корреляции	Критерий Фишера
$b_1 = -3.78197$ $b_0 = 1476.12$	165.189	28	$R^2 = -0.0227175$	$F = 0.000520331$

Таблица 5. Массив слов, сортированный по частотам

№	Количество
1	63
2	60
3	53
4	52
5	51
6	45

Заключение

Очевидно, что если в случайном тексте длины слов распределены равномерно, то никакого соответствия текста любому другому распределению не будет. С другой стороны, можно генерировать длины слов под заранее заданное распределение, что может выполнить любой робот. А тогда, какую проверку естественности происхождения текста можно выполнить по распределению?

На сегодня никто не доказал, что все языки подчиняются закону Ципфа, хотя возникает идея проверки этой гипотезы для всех языков представленных в Internete.

Опять-таки, а соответствует ли закону Ципфа текст написанный иероглифами? Я, например, очень в этом сомневаюсь; к тому же есть ещё и слоговое письмо.

В завершении заметим, что закон Ципфа не что иное, как распределение Пуассона при $\lambda = 1$.

Список использованных источников и литературы

1. Лапшин В. А. Лекции по математической лингвистике. – М. : Научный мир, 2010.
2. Википедия [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org> (Дата обращения: 07.02.2020).
3. Вэньтянь Ли, Закон Ципфа работает и для случайных текстов [Электронный ресурс]. – Режим доступа: <https://santafe.edu> (Дата обращения: 07.02.2020)
4. Норман Р. Дрейпер, Гарри Смит. Прикладной регрессионный анализ (третье издание). – М. : Диалектика, 2007.

List of references

1. Lapshin V. A. Lectures on mathematical linguistics, M, Scientific world, 2010.
2. Wikipedia, <https://ru.wikipedia.org>, accessed February 07, 2020.
3. Wentian Li, Zipf's Law also works for random texts, <https://santafe.edu>, accessed February 07, 2020.
4. Norman R. Draper, Harry Smith, Applied Regression Analysis (Third Edition). - M.: Dialectics, 2007.

МОТОВА А. С., РОЧЕВ К. В.
АВТОМАТИЗИРОВАННАЯ СИСТЕМА ВИРТУАЛЬНОГО
ПРОХОЖДЕНИЯ «ЭСКЕЙП РУМ» ДЛЯ СТУДИИ КВЕСТОВ УЗНИК
 УДК 004:794.5, ВАК 05.13.18, ГРНТИ 50.41.25

Автоматизированная система виртуального прохождения «Эскейп Рум» для студии квестов «Узник»

Automated system of virtual passage “Escape room” for the studio quest “Prisoner”

А. С. Мотова, К. В. Рочев

A. S. Motova, K. V. Rochev

Ухтинский государственный технический университет, г. Ухта

Ukhta State Technical University, Ukhta

В статье описаны работы по созданию автоматизированной системы виртуального прохождения «Эскейп Рум» для студии квестов «Узник». Данная автоматизированная система будет разработана в программе Unity на языке программирования C#.

The article describes the work on the creation of an automated system of virtual passage “Escape Room” for the studio quest “Prisoner”. This automated system will be developed in the Unity program in the C # programming language.

Ключевые слова: АС, квест, мобильное приложение

Keywords: AS, quest, mobile app

Введение

Квест – это увлекательная «живая» игра для команды из нескольких человек, в специально оборудованном для этого помещении. Участникам предлагается выполнить определенное задание, ограниченное по времени 60 минутами. В этом необычном и захватывающем приключении вам не обойтись без смекалки, логического мышления, эрудиции, а также ловкости, координации и умения взаимодействовать с товарищами.

Недавно, на волне популярности, новые комнаты-квесты начали появляться как грибы после дождя. Тогда у поклонников квестов, которых уже немало в нашей стране, встала проблема выбора. Это заставило организаторов задуматься о повышении качества предоставляемых услуг, в том числе о расширении ассортимента доступных сценариев. Так в квест-индустрии стали зарождаться жанры. На сегодняшний момент можно выделить 5 видов квестов, но в данной работе мы рассмотрим только один из них.

Эскейп-рум или как выбраться из комнаты. Знакомая практически всем ситуация: ваша команда заперта в помещении. Используя многочисленные подсказки и вспомогательные средства (которые еще необходимо найти), все должны за ограниченное время выбраться наружу. Обычно на это дается всего

один час. Открытие же, равно как проведение квестов сопряжено с рядом сложностей:

1) Инвестиции. Прежде чем квест начнёт приносить доход, в него нужно вложить много средств и сил. А именно – аренда помещения, постройка комнат, закуп необходимого оборудования.

2) Работники. Работа в квесте подразумевает постоянное общение с людьми всех возрастов, а, следовательно, вам надо найти таких сотрудников, которые смогут найти подход к каждому, и обучить их правильному проведению квеста.

3) Сценарий. Для наибольшего успеха сценарий должен быть интересным, по возможности разветвлённым, а самое главное уникальным. Причём основная проблема заключается в том, что для процветания бизнеса приходится 1–2 раза в год переписывать сценарий и переделывать квесты, чтобы интерес к ним не ослабевал.

Квестовые игры – развлечение не всегда безопасное. Если с квестами-комнатами все более-менее ясно, то при играх на городских улицах бывают и травмы, и несчастные случаи. Чтобы обезопасить и себя, и своих игроков от лишних проблем, организаторы либо меняют формат игры, полностью отказавшись от всех факторов риска, либо проводят инструктаж по технике безопасности, а на бумаге полностью отказываются от ответственности за действия игроков [1].

На основе вышеперечисленного разработка автоматизированной системы виртуального прохождения «Эскейп Рум» для студии квестов «Узник» является актуальной, так как она не требует огромного вложения средств, большого количества сотрудников, частого переписывания сценария, не имеет привязки к одному месту, а также будет более безопасным в отличии от квеста в реальности.

Предпроектное обследование

Рекламные игры – эффективный и перспективный рекламный метод. Рекламные игры помогают выстроить положительную коммуникацию потребителей с брендом и, главным образом, наладить стимулирование сбыта.

Одним из достоинств рекламных игр является то, что их нельзя отнести к прямым методам воздействия. То есть у потребителя, как правило, возникает желание поучаствовать в процессе, тем более что за это он может получить подарок.

Добиться отличного эффекта позволяет удачное сочетание рекламных игр и дарения продукции в качестве презента. Положительные эмоции очень важны, если вы хотите получить добровольные рекламные контакты.

Примерами таких игр являются:

1) Chevy Game Time – игра-викторина созданная для рекламы Супер Боула автомобильной компанией Chevrolet;

2) The Nightjar – игра-хоррор созданная для рекламы жевательной резинки 5ive [2].

Но, к сожалению, рекламные игры обладают одним недостатком – каждый раз их необходимо разрабатывать индивидуально под конкретную организацию. И случай со студией квестов «Узник» [3] не стал исключением.

Для её рекламы было решено реализовать аналог квеста в жанре «выход из комнаты» для мобильных устройств. Проведя опрос в группе студии квестов было выяснено, что более 70-ти % опрошенных пользуются телефонами на платформе Android, на основании этого для распространения приложения был выбран Play Market.

Средой разработки была выбрана Unity, так как она поддерживает все необходимые языки программирования, а также обладает интуитивно понятным интерфейсом и имеет бесплатную версию.

В соответствии с пожеланиями заказчика игра должна выполнять следующие функции (рис. 1):

- 1) формирование рейтинга игроков на основе статистики прохождения;
- 2) подсчет количества смертей игрока;
- 3) сбор данных пользователя для связи с ним;
- 4) возможность покупки подсказок для более быстрого прохождения;
- 5) реклама студии квестов «Узник» за счет ежемесячного конкурса;
- 6) наличие баннерной рекламы.

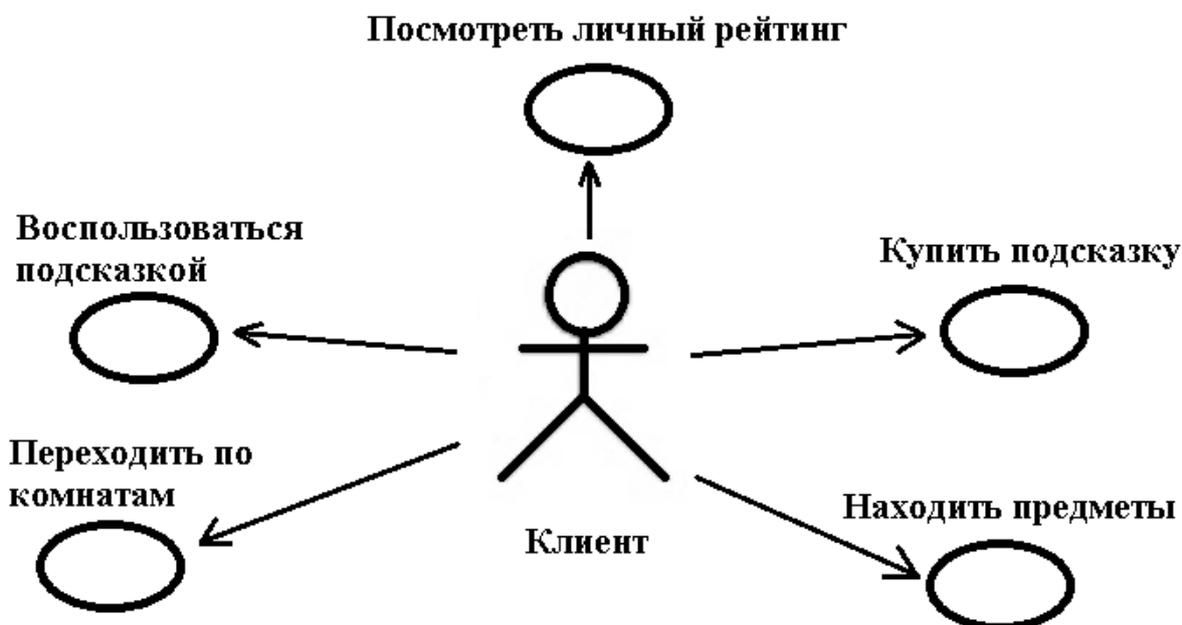


Рисунок 1. UML-диаграмма взаимодействия пользователя с системой

Проектирование информационной системы

На основе анализа предметной области была составлена следующая DFD-диаграмма «Как должно быть» – см. рисунок 2.

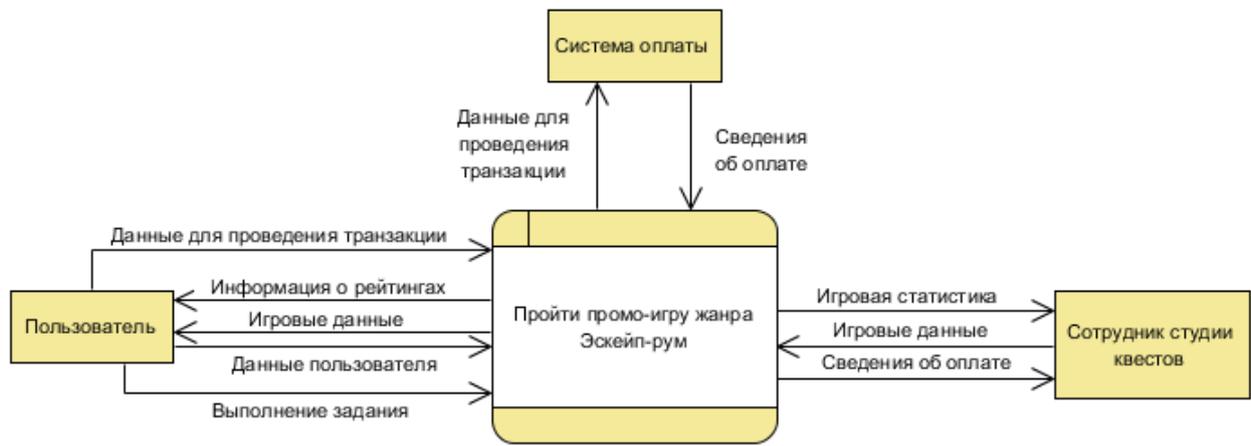


Рисунок 2. Контекстный уровень

После декомпозиции она приобрела вид, продемонстрированный на рисунке 3.

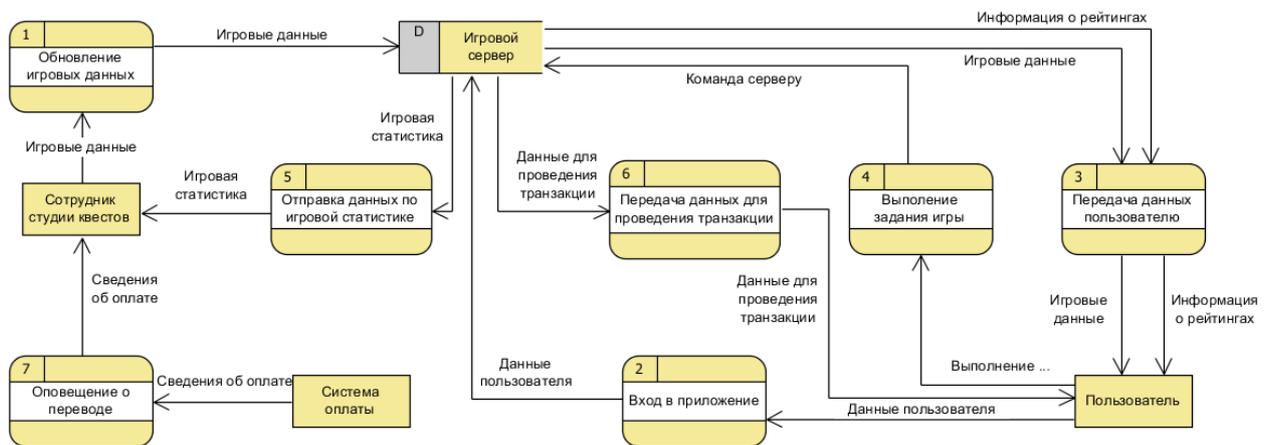


Рисунок 3. Системный уровень

На данных диаграммах можно выделить три сущности:

- 1) сотрудник студии квестов;
- 2) пользователь;
- 3) система оплаты

А также хранилище данных роль которого выполняет Игровой сервер
Каждая из сущностей выполняет свои функции:

Сотрудник обновляет данные на сервере и получает данные по игровой статистике.

Сервер передает пользователю следующие данные: данные о рейтинге, игровые данные или данные для проведения транзакции.

Пользователь отправляет на сервер свои данные или команду о выполняемом действии в головоломке.

Система оплаты оповещает сотрудника квеста о переводе денежных средств.

Для реализации связи с базой данных использовалась трехзвенная клиент-серверная архитектура. При такой схеме возможно централизованное администрирование прикладных функций и снижение сетевого трафика (т. к. с клиента передаются не sql-запросы, а всего лишь формы с необходимыми параметрами, которые затем используются на серверной части) (рис. 4).

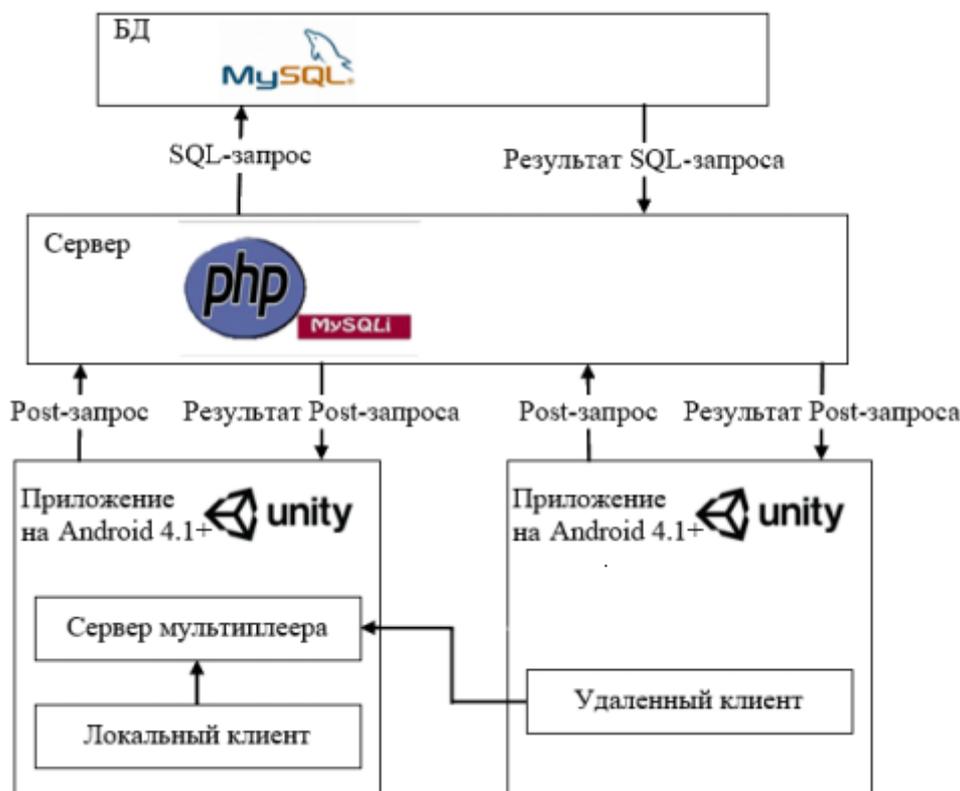


Рисунок 4. Архитектура системы

Результаты разработки системы

Результатом разработки игры стал почти готовый к эксплуатации программный продукт на игровом движке Unity [4], написанный на языке программирования C#.

Запустив игру, пользователь видит главное меню (рис. 5), где в настройках может регулировать громкость музыки, перейти на сайт студии квестов, а также внести данные о себе, а по кнопке «Убрать рекламу» сможет избавиться от баннерной рекламы.

Заключение

На основе анализа предметной области был разработан игровой продукт на платформе Android отвечающий основным требованиям заказчика:

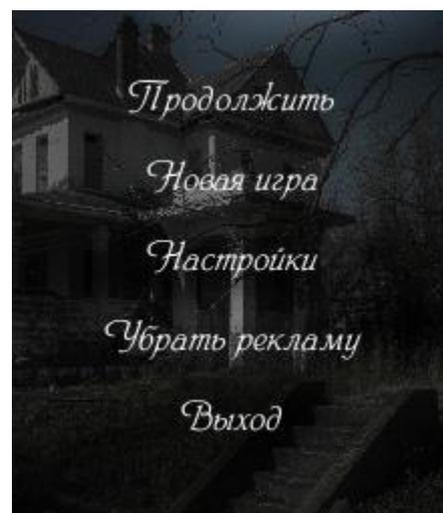


Рисунок 5. Главное меню

- 1) создан прототип с рабочим игровым процессом и интерфейсом;
- 2) создана база данных и реализованы все связанные с ней функции;
- 3) осуществлено портирование на целевые платформы.

Помимо игровой сцены было реализовано меню с необходимыми подразделениями, имеющее эргономичный дизайн с квестовой тематикой.

Список использованных источников и литературы

1. Бизнесменс [Электронный ресурс]. URL: <http://businessmens.ru/article/idei/ideya-biznesa-kak-otkryt-kvest--igru-v-real-nosti> (дата обращения 14.02.2019)
2. Канобу [Электронный ресурс]. URL: <https://kanobu.ru/articles/igraj-i-potrebyaj-2-10-reklamnyih-igr-dlya-mobilnyih-telefonov-338394/> (дата обращения 14.02.2019)
3. Сайт студии квестов «Узник» [Электронный ресурс]. URL: <http://kvest11.ru/> (дата обращения 14.02.2019)
4. Сайт игрового движка и средства для разработки игр Unity [Электронный ресурс]. URL: <https://unity3d.com/ru/> (дата обращения 17.03.2019).

List of references

1. Businessmens, <http://businessmens.ru/article/idei/ideya-biznesa-kak-otkryt-kvest--igru-v-real-nosti>, accessed February, 14, 2019.
2. Kanobu, <https://kanobu.ru/articles/igraj-i-potrebyaj-2-10-reklamnyih-igr-dlya-mobilnyih-telefonov-338394/>, accessed February, 14, 2019.
3. Studio quest PRISONER, <http://kvest11.ru/>, accessed February, 14, 2019.
4. Official website of game engine and game development platform Unity, <https://unity3d.com/ru/>, accessed March, 17, 2019.

**КИМ М. А., ДОРОГОБЕД А. Н.
ИНФОРМАЦИОННАЯ СИСТЕМА «ЧИР СПОРТ РЕСПУБЛИКИ
КОМИ»: ПОДСИСТЕМА УЧАСТНИКОВ ФЕДЕРАЦИИ И
ПОКАЗАТЕЛЕЙ ИХ ДЕЯТЕЛЬНОСТИ**

УДК 658.512, ВАК 05.13.01, ГРНТИ 50.49.33

Информационная система «Чир спорт
Республики Коми»: подсистема
участников федерации и показателей
их деятельности

Information system “Chir Sport of
the Komi Republic”: a subsystem of
the participants of the federation and
indicators of their activities

М. А. Ким, А. Н. Дорогобед

M. A. Kim, A. N. Dorogobed

Ухтинский государственный
технический университет, г. Ухта

Ukhta State Technical University,
Ukhta

В статье рассматривается информационная система «Чир спорт Республики Коми»: подсистема участников федерации СЧСЧР РК и показателей их деятельности. В ходе изучения предметной области были выявлены следующие проблемы: потеря информации, отсутствие централизованного центра хранения информации в электронном виде; ошибки при занесении информации. Предлагаемая система позволит снизить трудозатраты учетного персонала организации за счёт уменьшения объема бумажной работы. Особое внимание уделяется исследованию предметной области, моделированию системы, выбору архитектуры системы и ее реализации.

The article deals with the information system "Chir sport of the Republic of Komi": a subsystem of the indicators of activities. During the study of the subject area, the following problems were identified: loss of information, the lack of a centralized center for storing information in electronic form; errors in entering information. The proposed system will reduce the labor costs of the accounting staff of the organization by reducing the amount of paperwork. Particular attention is paid to the study of the subject area, system modeling, the choice of system architecture and its implementation.

Ключевые слова: информационная система, союз чир спорта и черлидинга, учёт

Key words: information system, union of sports and cheerleading, accounting

Введение

Структурное подразделение Союза Чир Спорта и Черлидинга России «Союз чир спорта и черлидинга Республики Коми» – спортивная организация, объединяющая клубы, команды и спортивные соревнования, которая была создана в

2014 года. С момента создания отделения, несмотря на короткий срок его существования, прослеживается тенденция по увеличению численности занимающихся чир спортом в спортивных секциях и группах физкультурно-оздоровительной направленности.

Создание информационного обеспечения процесса учёта участников спортивной организации, позволит сделать процесс подготовки отчётности прямым результатом учёта, а также усовершенствовать процесс учёта таким образом, чтобы максимально снизить объем ручного ввода. Система включает в себя две подсистемы: подсистема «Учёт соревнований», которая предназначена для формирования отчётности по соревнованиям, учёту самих соревнований и регистрации участников, подсистема «Учёт пользователей» отвечает за регистрацию участников коллектива, коллективов, организаторов и судей.

Предпроектное обследование

Основными процессами в работе организации являются учёт участников коллектива, клубов, организаторов, судей и формирование отчётности в головной офис. Сущность учёта, состоит в наблюдении за развитием путем регистрации данных участников, их обработке и хранении полученной информации.

В процессах спортивной организации принимают роли: руководитель клуба, организатор соревнований, региональное руководство отделения СЧСЧР в РК и судья.

Руководитель клуба – представитель танцевального коллектива, который несёт ответственность за каждого участника, имеет право на просмотр, добавление и редактирование карточки клуба, получение различных выходных форм и отчётов.

Организатор соревнований – просматривает события, которые он провел или ведёт, редактирование личной информации.

Судья – заполняет информацию о себе, после чего создается карточка спортивного судьи, имеет возможность редактирования судейской карточки, формирование карты спортивного судьи за заданный период.

Основываясь на приведенных, в общем, и конкретном описаниях процесса и видении его проблем были определены границы системы.

На рисунке ниже (рис. 1) представлена контекстная диаграмма потоков данных.

Результатом декомпозиции процесса учёта, которой стала диаграмма потоков данных первого уровня (рис. 2). Полученная на рисунке диаграмма демонстрирует особенности поведения информационной системы.

Система должна обладать следующим набором функций:

- учёт участников коллектива;
- учёт клубов;
- учёт судей;
- учёт организаторов соревнований;
- возможность подтверждения взносов;
- формирование отчётности.

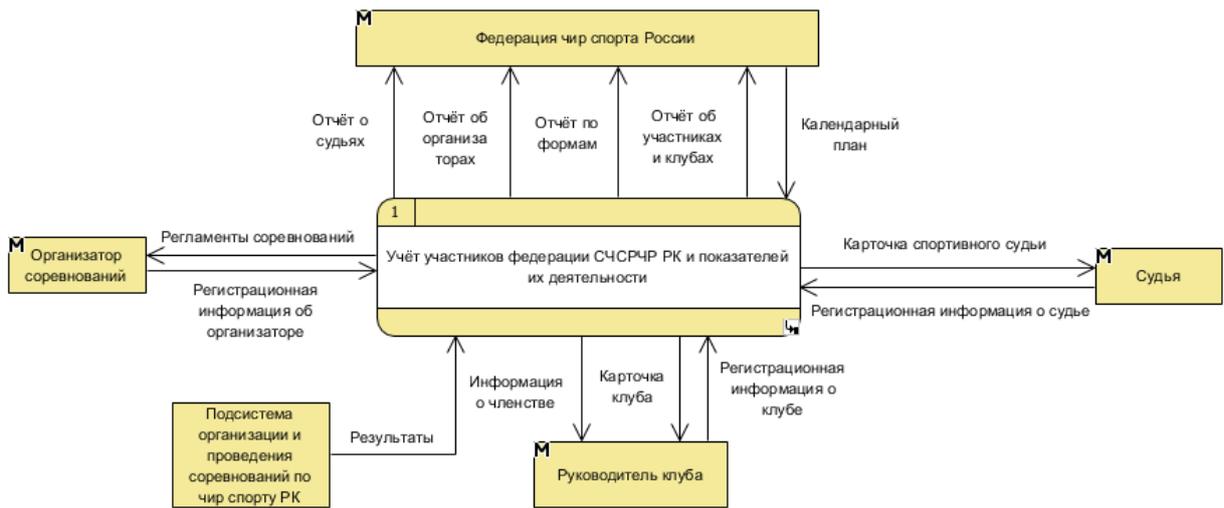


Рисунок 1. Контекстная диаграмма

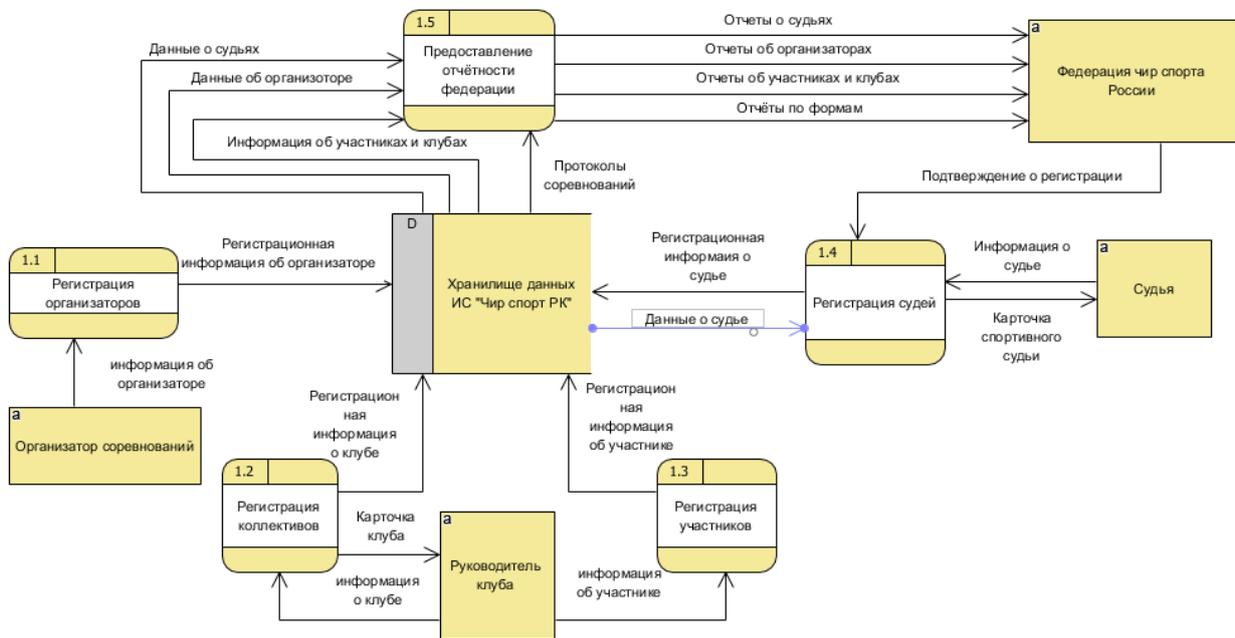


Рисунок 2. Декомпозиция первого уровня

Подсистема учёта участников федерации СЧСЧР РК разрабатывается для широкого круга лиц, которые должны иметь возможность войти в систему с любого устройства. Для минимизации затрат ресурсов на адаптацию системы под различные устройства и её внедрение было решено разработать веб-приложение.

Веб-приложение реализует «клиент-сервер» трехзвенную архитектуру, в которой: клиент – браузер, веб-сервер и сервер базы данных. Клиентская часть представляет собой пользовательский интерфейс, откуда формируются запросы к серверу, и идет обработка ответов от сервера.

Серверная часть получает и обрабатывает запросы от пользователя, в соответствии с запросом отправляет свой запрос на сервер базы данных, получая в ответ данные из БД, и формирует веб-страницу, которую отправляет клиенту по сети с помощью протокола HTTP.

для отображения веб-страниц, чтобы воспользоваться разрабатываемым веб-приложением.

В результате проведенной работы были выделены основные сущности, их атрибуты и взаимосвязи между сущностями, на основании чего построена логическая модель БД.

Заключение

На данный момент проведено изучение предметной области, была построена концептуальная модель процесса, показывающая внешние сущности, с которыми взаимодействует процесс. В дальнейшем диаграмма декомпозирована до модели потоков данных первого уровня. На основе моделей и рассмотренной предметной области были выявлены проблемы учёта. Выбрана архитектура системы и построена модель БД.

Список использованных источников и литературы

1. Сайт федерации cheerliding.su/ [Электронный ресурс]. URL: <http://www.cheerleading.su/> (дата обращения 23.05.2019)
2. Сайт организации ortodance.ru/ [Электронный ресурс]. URL: <http://www.ortodance.ru/> (дата обращения 23.05.2019)
3. Допира Р. И., Попова Н. В. Технология ASP.NET MVC // Молодой ученый. 2018. №49. С. 17–20. URL <https://moluch.ru/archive/235/54577/> (дата обращения: 06.06.2019).
4. Научная электронная библиотека eLIBRARY.RU <https://elibrary.ru/defaultx.asp>
5. Раздел поддержки официального сайта CASE–средства Visual Paradigm [Электронный ресурс]. URL: <https://www.visualparadigm.com/support/> (дата обращения: 07.06.2019).

List of references

1. Federation site cheerliding.su, <http://www.cheerleading.su/>, accessed May 23, 2019.
2. Organization website ortodance.ru, <http://www.ortodance.ru/>, accessed May 23, 2019.
3. Dopira, R. I., Popova, N. V., “ASP.NET MVC Technology”, *Young Scientist*, 2018, no. 49, pp. 17–20, <https://moluch.ru/archive/235/54577/>, accessed 06.06.2019.
4. Scientific electronic library eLIBRARY.RU, <https://elibrary.ru/defaultx.asp>.
5. The support section of the official site CASE – Visual Paradigm, <https://www.visualparadigm.com/support/>, accessed 07/06/2019.

СВЕДЕНИЯ ОБ АВТОРАХ

Базарова Ирина Александровна

Ухтинский государственный
технический университет, г. Ухта;
доцент кафедры ВТИСиТ, доцент

Bazarova Irina Aleksandrovna

Ukhta State Technical University,
Ukhta, Associate professor,
Department of computer science,
information systems and technologies

E-mail: ibazarova@ugtu.net

Бильфельд Николай Валентинович

Березниковский филиал «Пермского
национального исследовательского
политехнического университет»,
г. Березники; кандидат технических
наук, доцент, доцент

Bilfeld Nikolay Valentinovich

Bereznikovsky branch of the Perm
National Research Polytechnic
University, Berezniki;
Candidate of Technical Sciences,
associate professor

E-mail: bilfeld@mail.ru

Володина Юлия Игоревна

Березниковский филиал «Пермского
национального исследовательского
политехнического университет»,
г. Березники; кандидат технических
наук, доцент

Volodina Julia Igorevna

Bereznikovsky branch of the Perm
National Research Polytechnic
University, Berezniki;
Candidate of Technical Sciences,
associate professor

E-mail: julia_volodina@mail.ru

Гатин Герман Николаевич

Ухтинский государственный
технический университет, г. Ухта;
доцент кафедры вычислительной
техники, информационных систем и
технологий, доцент

Gatin German Nikolaevich

Ukhta State Technical University,
Ukhta;
Associate professor, Department of
computer science, information systems
and technologies

Дорогобед Алёна Николаевна

Ухтинский государственный
технический университет, г. Ухта;
кандидат технических наук,
и. о. заведующего кафедрой
вычислительной техники,
информационных систем и
технологий

Dorogobed Alyona Nikolaevna

Ukhta State Technical University,
Ukhta;
Candidate of technical sciences,
Acting Head of the Chair Department
of computer science, information
systems and technologies

E-mail: lekun90@mail.ru

Ким Милана Алексеевна

Ухтинский государственный
технический университет, г. Ухта;
студент кафедры вычислительной
техники, информационных систем и
технологий

Kim Milana Alekseevna

Ukhta State Technical University,
Ukhta; Student, Department of
computer science, information systems
and technologies

Мотова Анастасия Станиславовна

Ухтинский государственный
технический университет, г. Ухта;
студент кафедры вычислительной
техники, информационных систем и
технологий

Motova Anastasia Stanislavovna

Ukhta State Technical University,
Ukhta; Student, Department of
computer science, information systems
and technologies

Поторопин Вячеслав Алексеевич

Ухтинский государственный
технический университет, г. Ухта;
студент кафедры вычислительной
техники, информационных систем и
технологий

Potoropin Vyacheslav Alekseevich

Ukhta State Technical University,
Ukhta; Student, Department of
computer science, information systems
and technologies

Рочев Константин Васильевич

Ухтинский государственный
технический университет, г. Ухта;
кандидат экономических наук,
доцент кафедры вычислительной
техники, информационных систем и
технологий

Rochev Konstantin Vasilievich

Ukhta State Technical University,
Ukhta;
candidate of economic sciences,
Associate professor, Department of
computer science, information systems
and technologies

E-mail: k@rochev.ru

Ухтинский государственный технический университет

Информационные технологии
в управлении и экономике
2020, № 01

Information technology
in management and economics
2020, No 01

ISSN 2225-2819

Свидетельство о регистрации Эл. № ФС77-65216

Адрес редакции: 169300, г. Ухта, ул. Первомайская, 13

Интернет-сайт: <http://it-ugtu.ru>, <http://itue.ru/>, <http://итгуэ.рф>

Электронная почта: info@itue.ru

Телефон: 8 (8216) 700-308

Главный редактор: *К. В. Рочев*

Дизайн и компьютерная вёрстка: *А. В. Семяшкина*