УДК 62-503.57

ЗАДАЧА ПРОГРАММНОЙ РЕАЛИЗАЦИИ АВТОМАТИЗИРОВАННОГО ГЕНЕРАТОРА СИСТЕМНО ОБОСНОВАННОГО ТЕХНИЧЕСКОГО ЗАДАНИЯ НА ИНФОРМАЦИОННЫЕ СИСТЕМЫ

Фирстков А.Л.[1], Гольдштейн С.Л.[1], Грицюк Е.М.[2]

[1]ФГАОУ ВО «УрФУ имени первого Президента России Б.Н.Ельцина», г. Екатеринбург, Россия

[2]ГАУЗ СО МКМЦ «Бонум», г. Екатеринбург, Россия

Резюме. В статье рассмотрена задача программной реализации автоматизированного генератора системно обоснованного технического задания (АГ СО ТЗ) на информационные системы, в частности, медицинские. Для её решения предложено развитие алгоритма работы АГ СО ТЗ, найдены аналоги одной из незавершённых систем, произведён их обзор и составлены пакеты прототипов. В условиях конкуренции предложено обратить внимание на проблему утечки информации о техническом задании.

Ключевые слова: техническое задание, программное обеспечение, инженерная реализация, разработка, информационная система

The problem of program implementation automated generator system-defined technical assignment

Firstkov A.L.[1], Goldstein S.L.[1], Gritsyuk E.M.[2]

[1] Yeltsin UrFU or UrFU, Ekaterinburg, Russia

[2]State Autonomic Health Institution of the Sverdlovsk Region Multiprofile Clinical Medical Center «BONUM»

The task of software implementation of the automated generator of the systematically justified technical task (AG SO TZ) on information systems, in particular, medical ones, is considered in the article. To solve this problem, we propose the development of the algorithm for the work of AG TS TZ, find analogs of one of the incomplete systems, review them and compile prototype packages. In the conditions of competition, it is suggested that attention be paid to the problem of leakage of information about the technical task.

Keywords: technical specification, software, engineering implementation, development, information system.

Введение

Как известно, техническое задание (ТЗ) необходимо для разработки качественного программного обеспечения (ПО), на всех стадиях жизненного цикла ПО особенно с ростом его сложности. Грамотно составленное ТЗ будет полезно. Отсутствие этого документа может привести к тому, что даже при относительно низкой сложности ПО в случае возникновения необходимости внесения изменений или небольшой доработки проще будет начать разработку «с нуля». Существующие ПО для разработки ТЗ имеют ряд недостатков, в том числе высокую цену [1], и отсутствие настройки на специфику предметной области, например, медицинской организации, излагаемую в специальных нормативных документах [2]. Автоматизированный генератор системно обоснованного технического задания (АГ СО ТЗ) [3] – это автоматизированная многопользовательская информационная система (ИС), за время работы над которой (более 5 лет), накоплено большое количество идей и методов разработки. Реализация её программной части – важная задача, которую необходимо решить для эффективной работы системы (далее под генератором будем понимать программную часть ИС, если не оговорено иное). В таблице 1 приведена иерархия структурных элементов генератора.

Таблица 1
Иерархия АГ СО ТЗ

Уровень	Название уровня	Объект	Структуры, рассмотренные в статье
0	Генератор	ПО информационной системы	Прототип
1	Система	Функционально выделенная часть	Система шаблонов
2	Подсистема	Совокупность связанных модулей с общими входами и выходами	Подсистемы каталогизации, оценки качества,
3	Модуль	Несколько связанных по смыслу функций	Модуль размещения компонентов, размещения текста
4	Блок	Элемент, выполняющий конкретную функцию	В составе соответствующих модулей

Алгоритмы работы генератора неточны, недостаточно подробны и актуальны. Для дальнейшего развития его прототипа необходимо доработать существующий или составить новый алгоритм. В условиях конкуренции ПО для разработки ТЗ должно способствовать сохранению коммерческой тайны. Необходимо предусмотреть это в алгоритме. В предыдущей публикации [4] мы пришли к выводу, что для функционирования АГ СО ТЗ требуется доработать и запрограммировать систему шаблонов с последующей интеграцией её с генератором. Также у прототипа отсутствует стандарт оформления кода. В статье поставлена и решена задача развития алгоритма работы генератора, обзора аналогов системы шаблонов, выбора из них прототипов,

выявления недостатков у корпоративного прототипа (генератора) и преодоления недостатков.

Поиск и обзор аналогов алгоритма работы АГ СО ТЗ

Алгоритмы работы АГ СО ТЗ публиковались в электронном научном журнале «Системная интеграция в здравоохранении». В результате поиска в журнале найдено несколько вариантов моделей функционирования АГ СО ТЗ [5, 6]. Один из них [6] предлагает в дополнение создания ТЗ по существующей технологии в случае возникновения сложной ситуации при разработке ТЗ использовать системы настройки на специфику и интеграции. Другой алгоритм [5] подробно описывает работу генератора и систем, поэтому в качестве прототипа выбран наиболее полно изложенный аналог [5].

Предлагаемый алгоритм АГ СО ТЗ

При анализе прототипа [5] выяснили, что он не обладает достаточной гибкостью: пользователь в любой момент времени может начать работу с любой системой, в алгоритме [5] это сделать невозможно. А для того, что бы действия пользователя при такой гибкости имели смысл, перед выбором системы необходима работа блока анализа ситуации. В случае если ТЗ в конце одной из итераций цикла по ресурсам не удовлетворяет качеству, алгоритм [5] предлагает вернуться в самое начало, не повторяя только ввод. Это приведёт к тому, что работу над ТЗ придётся начать сначала, не разрешив при этом проблемную ситуацию, что в результате может повторно многократно приводить к той же проблеме. Также детально не представлена работа пользователя над ТЗ и его взаимодействие с генератором.

Для решения выявленных проблем модернизирован алгоритм, взятый за прототип и приведённый рисунке 1. Системы, обозначенные индексом k: 1 - электронизации входной информации, 2 – работы с заказчиком и разработчиком, 3 - фабрики моделей, 5 - шаблонов, 6 – интегратора ТЗ, 7 - управления, 8 - визуализации, 9 – печати, 10 - настройки на специфику медико-инженерного объекта и вида медицинской деятельности, 11 – системной интеграции. Вновь вводимые блоки, обозначенные на рисунке 1 номерами: 9 – анализа ситуации, 39 – анализа проблем.

В предложенном алгоритме модифицирован блок оценки качества (37), не обозначенный в алгоритме прототипе [5] (но его наличие подразумевалось перед блоком 9.33). Перед началом работы с выбранной системой пользователь может с помощью репозитария обратиться к данным, расположенным: локально – для каталогизации и извлечения во время работы и глобально – для сохранения, загрузки и поддержки актуальности общедоступных данных.

Подробное описание взаимодействия с репозитарием (обозначен 4 на рисунке 1) необходимо для разработки многопользовательской версии генератора. В условиях конкуренции для сохранения коммерческой тайны, после завершения сеанса работы пользователь отправляет все данные из локального репозитария в хорошо защищённый глобальный.

На вход генератор получает задание на разработку ТЗ, документы, данные об объекте компьютеризации и аналоги ТЗ. Работа над ТЗ организована в цикле по ситуациям, работа в каждой из которых организована по ресурсам. Загрузив при необходимости данные из репозитария, пользователь анализирует ситуацию и выбирает одну из систем генератора обозначенных индексом k, с которой будет работать. Пользователь и выбранная им система работают параллельно. Система осуществляет информационную поддержку пользователя (пользователь отправляет запрос системе) и выполняет работу, которая может быть автоматизирована (если работа не может быть автоматизирована, система отправляет запрос пользователю). Работа пользователя с системой завершается, когда пользователь и система завершили работу и ответили на запросы друг друга.

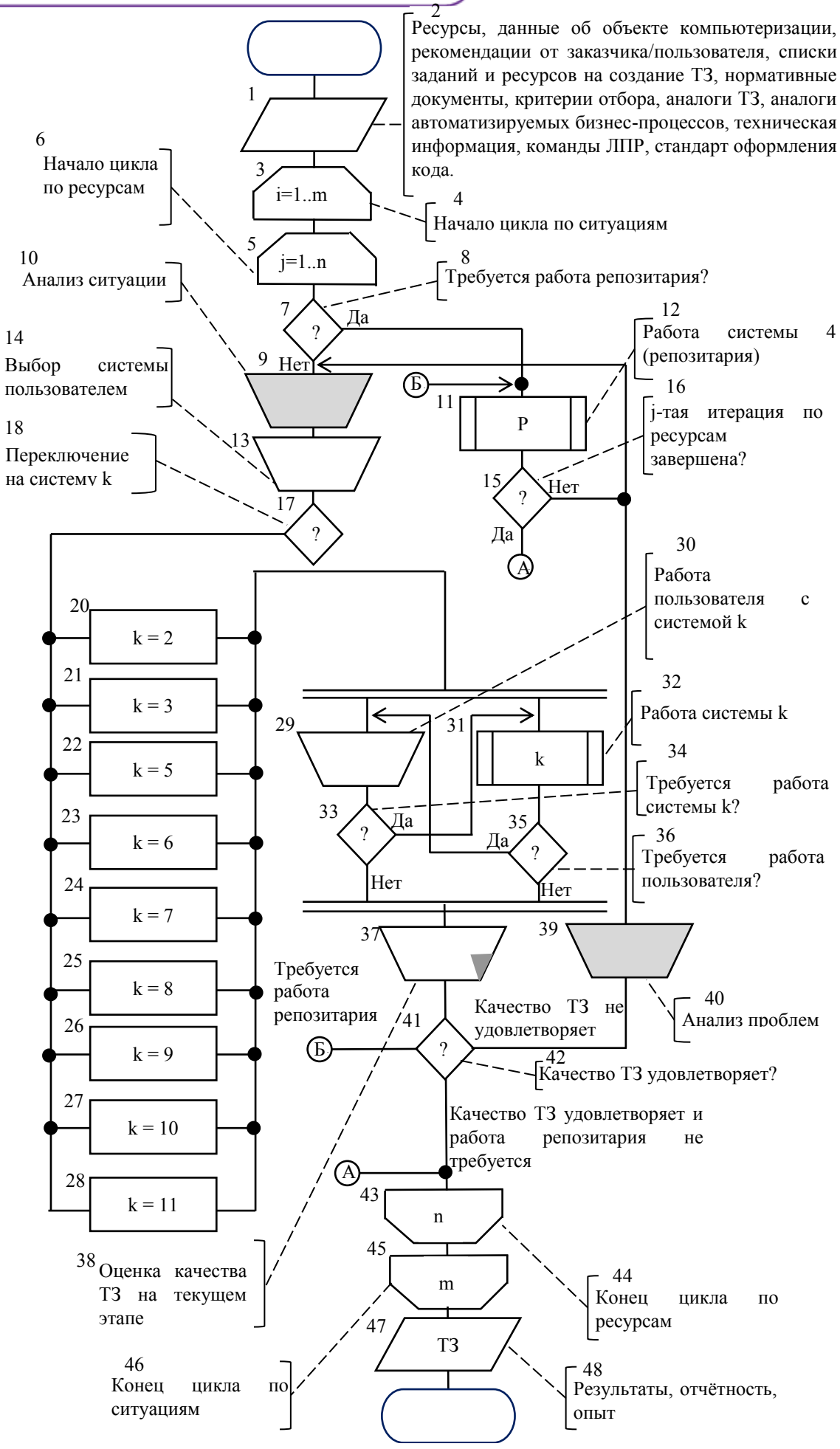


Рис.1 Алгоритм работы АГ СО ТЗ

Литературный обзор и поиск аналогов системы шаблонов

Следующая задача, решение которой необходимо для развития генератора, рассмотренная в статье – разработка системы шаблонов. Система шаблонов ранее разрабатывалась [4], [7], но мы пришли к выводу, что целесообразно учитывая предыдущий опыт вернуться к поиску и обзору аналогов.

Источники информации можно разделить на 3 группы:

1) интернет: были использованы браузеры Google Chrome и Yandex, а также официальные сайты и профессиональные порталы (например, Хабрахабр);
2) печатные издания: в основном публикации в электронном научном журнале «Системная интеграция в здравоохранении»;
3) эксперты: опрос и консультации у профессорско-преподавательского состава УрФУ и сотрудников МКМЦ «Бонум».

Для улучшения функционирования АГ СО ТЗ рассмотрена программная реализация его запатентованного корпоративного прототипа [3], на который также получено 2 свидетельства на программный продукт [8,9]. В качестве аналогов системы шаблонов выбраны ПО или сервисы, позволяющие осуществлять работу с программной документацией [10-14]. При составлении пакета прототипов для более полной детализации было найдено ещё 16 узкоспециализированных аналогов [15-30]. Также были проанализированы различные стандарты и рекомендации оформления кода на c# [31-36].

Работа с аналогами

При работе с аналогами пришлось ужесточить правила отбора и исключить часть из них [13]. Аналог [13] Requirements Miner for AllFusion Modelling Suite позволяет автоматически создавать ТЗ, но делает это импортируя данные из другого ПО, поэтому не обладает достаточно большим количеством функций, рассматриваемых в таблице 2 и не участвует в сравнении. По своим функциям [13] является удачным прототипом для другой системы АГ СО ТЗ – интегратора.

Для сравнения аналогов составлены таблицы, в которых предполагаемые элементы разрабатываемой системы оцениваются у рассматриваемого аналога, числами от 0 до 1 включительно. Аналог, получивший наибольшую сумму оценок элементов, отбирается в пакет прототипов. Аналоги, получившие оценки, отличающиеся от наибольшей менее чем на 0.5, также входят в пакет прототипов. Если элемент разрабатываемой системы присутствует у небольшого числа аналогов или получает невысокую наибольшую оценку, то для данного элемента составляется отдельная таблица с добавлением новых аналогов.

В таблице 2 аналог [10] Author IT получил высокие оценки за формирование запроса к базе данных (БД), так как организует работу с данными с помощью встроенной системы управления базами данных (СУБД). В аналогах [11] TZGen, [14] Мастер ТЗ компоненты для ввода данных размещены на формах (разделах ТЗ) без возможности добавления, удаления и перераспределения. У аналогов [10], [12] Help+Manual документ имеет редактируемую иерархическую структуру, между разделами которой можно распределять данные. В рассмотренных аналогах отсутствуют функции оценки качества шаблона и стандартизации оформления кода. Исходя из оценок, приведённых в таблице 2, прототипом системы шаблонов выбран аналог Author IT [10], а модуля разделения на ЭФ аналоги Author IT [10] и Help+Manual [12].

Таблица 2
Аналоги системы шаблонов

Критерии сравнения		Оценки аналогов по источникам информации			
Подсистема	Модуль	[10]	[11]	[12]	[14]
5.1 содержания шаблона	5.1.1 разделения на ЭФ	0.9	0.4	1	0.3
	5.1.2 размещения компонентов	0.6	0	0.3	0
	5.1.3 размещения текста	0.9	0.3	0.7	0.2
5.2 каталогизации	5.2.1 ведения каталогов входных, промежуточных и выходных файлов	0.8	0.1	0.7	0.2
	5.2.2 обзора документов	0.6	0	0.6	0.2
	5.2.3 взаимодействия с репозитарием	0.7	0.1	0	0
5.3 формирования запроса к базе данных	5.3.1 создания таблиц	1	0	0	0
	5.3.2 создания полей	1	0	0	0
	5.3.3 создания связей и ограничений	1	0	0	0
5.4 оценки качества шаблона	5.4.1 измерения эталонных значений свойств	0	0	0	0
	5.4.2 измерения свойств исследуемого образца	0	0	0	0
	5.4.3 сопоставления полученных результатов с эталонными значениями	0	0	0	0
5.5 стандартизации оформления кода	5.5.1 парсинга кода	0	0	0	0
	5.5.2 форматирования кода	0	0	0	0
	5.5.3 настройки форматирования	0	0	0	0
Итого		7.5	0.9	3.3	0.9

Из таблицы 2 следует, что аналоги системы шаблонов не идеально выполняют её предполагаемые функции (а в случае с оценкой качества и стандартизацией кода фактически их не выполняют). Для более глубокого анализа и обоснования данных экспертных оценок рассмотрены аналоги для каждой из подсистем системы шаблонов, кроме подсистем содержания шаблона, для которой рассматриваются аналоги её модулей. В результате в базу аналогов добавлены: ПО для каталогизации [15-17], облачные хранилища данных [18-20], сервисы для оценки качества текста [21-23], сервисы для построения экранных форм (ЭФ) [24-26], текстовый редактор [27], приложения для разработки ПО [28-30], стандарты оформления кода на c# [31-36].

Работа с компонентами в [27] Microsoft Word доступна с помощью VBA. Так как нельзя требовать от пользователя знаний и навыков объектно-ориентированного программирования оценки за функции модуля размещения компонентов для [27] в таблице 3 существенно занижены. Аналоги [24-26] получили очень близкие оценки, так как их функции и возможности слабо различаются. При этом один из них [24] Google Формы получил наибольшую оценку. Прототипами модуля размещения компонентов выбраны аналоги [24-26].

Таблица 3
Аналоги модуля размещения компонентов

Критерии сравнения		Оценки аналогов по источникам информации			
Блок	Характеристики блоков	[24]	[25]	[26]	[27]
5.1.2.1 размещения компонентов	Удобство интерфейса	0.9	0.8	0.7	0.7
	Возможности размещения компонентов на форме и её настройки	0.9	0.9	0.9	0.4
5.1.2.2 создания компонента	Разнообразие набора компонентов	1	1	1	0.3
	Возможности настройки	0.6	0.6	0.6	0.3
Итоговая оценка		3.4	3.3	3.2	1.7

Аналог [27] на высоком уровне реализует все функции модуля размещения текста, другой аналог [12] уступает в возможностях работы с формулами и незначительно ввода текста. Исходя из оценок, приведённых в таблице 4, прототипами модуля размещения текста выбраны аналоги [27], [12].

Таблица 4
Аналоги модуля размещения текста

Критерии сравнения		Оценки аналогов по источникам информации		
Блок	Характеристики блоков	[10]	[12]	[27]
5.1.2.1 размещения текста	Удобство размещения и ввода	0.7	0.9	1
	Возможности ввода специальных символов	0.8	0.8	1
5.1.2.2 оформления текста	Возможности настройки стилей текста	0.7	1	1
	Возможности настройки начертания символов	0.8	1	1
Итоговая оценка		3	3.7	4

Аналоги [15-17] выполняют функции ведения каталогов на достаточно высоком уровне. При сортировке и поиске большее количество параметров позволяют учесть аналоги [15] Where Is It? и [16] WinCatalog 2017. Но рассмотренные в таблице 5 ПО не предусматривают работу с удалённым хранилищем данных, поэтому для модуля взаимодействия с репозитарием произведены поиск и оценка аналогов, приведённая в

таблице 6. По результатам оценок, приведённых в таблице 5, прототипами подсистемы каталогизации выбраны аналоги [15] и [16].

Таблица 5
Аналоги подсистемы каталогизации

Критерии сравнения		Оценки аналогов по источникам информации		
Модуль	Блок	[15]	[16]	[17]
5.2.1 ведения каталогов входных, промежуточных и выходных файлов	5.2.1.1 импорта файлов	1	1	0.7
	5.2.1.2 работы с директориями	1	1	1
	5.2.1.3 ведения каталогов	1	0.9	0.9
5.2.2 обзора документов	5.2.2.1 сортировки файлов и каталогов	1	0.9	0.4
	5.2.2.2 поиска файлов	1	0.8	0.6
5.2.3 взаимодействия с репозитарием	Добавления	0	0	0
	Загрузки	0	0	0
	Структуризации	0.7	0.7	0.8
Итоговая оценка		5.7	5.4	4.4

В таблице 6 оценивается клиентская часть облачных хранилищ, поскольку требуется найти аналоги для модуля взаимодействия, а не самого репозитария. У рассмотренных аналогов требуемые для модуля функции реализованы на достаточно высоком уровне. Все аналоги позволяют выполнять поиск файла, проверку файла, например, с помощью вычисления контрольной суммы и собирают метаданные, такие как дата последних изменений, владелец, дата последнего обращения к файлу. Аналог [20] GitHub предоставляет большие возможности по управлению версиями, в то время как аналоги [18] Google Диск и [19] Dropbox позволяют просто хранить несколько файлов, обозначенных, как разные версии одного файла. По результатам оценок, приведённых в таблице 6, прототипами модуля взаимодействия с репозитарием выбраны аналоги [18], [19] и [20].

Таблица 6
Аналоги модуля взаимодействия с репозитарием

Критерии сравнения		Оценки аналогов по источникам информации		
Блок	Характеристики блоков	[18]	[19]	[20]
5.2.3.1 добавления	Распределение файлов	0.8	1	1
	Аутентификация	1	1	1
5.2.3.2 загрузки	Проверка файла	1	1	1
	Поиск файла	1	1	1
5.2.3.3 структуризации	Сбор метаданных	1	1	1
	Управление версиями	0.7	0.7	1
Итоговая оценка		5.5	5.7	6

Интегрированная в аналог [10] СУБД полностью выполняет функции хранения данных, но имеет закрытую архитектуру, что препятствует взаимодействию с другим ПО и делает

невозможным модернизацию и оценку качества работы. Создание запроса к БД в аналогах [28-31] автоматизировано, но не может выполняться без участия пользователя. ПО [28-29] предоставляет возможности по настройке и проектированию запроса, которые лучше реализованы у более узкоспециализированного аналога [29] Microsoft Access. По результатам оценок, приведённых в таблице 7, прототипами подсистемы формирования запроса к базе данных выбраны аналоги [10], [28-29].

Таблица 7

Аналоги подсистемы формирования запроса к базе данных

Критерии сравнения		Оценки аналогов по источникам информации		
Модуль	Блок	[10]	[28]	[29]
5.3.1 создания таблиц	5.3.1.1 разделения данных по таблицам	0.9	0.9	0.9
	5.3.1.2 создания первичного ключа	1	1	0.9
5.3.2 создания полей	5.3.2.1 выбора типа данных	0.9	0.9	0.8
	5.3.2.2 настройки свойств поля	0.8	0.7	1
5.3.3 создания связей и ограничений	5.3.3.1 создания связей	0.8	0.8	0.9
	5.3.3.2 настройки обеспечения целостности данных	1	0.9	0.9
Итоговая оценка		5.4	5.2	5.4

Из аналогов [21-23] только один [21] WinMerge 2.14.0 даёт возможность вводить эталонные параметры, у аналога [23] TopWriter эта функция реализована как ввод текстов для сравнения, а сервис [22] Istio предоставляет выбор большого количества параметров для анализа, что компенсирует отсутствие загрузки пользовательских критериев. Аналог [23] позволяет загружать тексты для сравнения, длина которых не превышает 5000 символов для сравнения и 1000 для анализа (возможно, этот недостаток будет частично или полностью устранён), поэтому его возможности загрузки эталонного образца уступают сервисам [21-22]. Аналоги [21-22] в удобной форме отображают большое количество параметров, в отличие от сервиса [23], который в качестве результата выводит процент схожести текстов либо предложения по изменению текста. По результатам оценок, приведённых в таблице 8, прототипом подсистемы оценки качества выбран аналог [21].

Таблица 8
Аналоги подсистемы оценки качества

Критерии сравнения		Оценки аналогов по источникам информации		
Модуль	Блок	[21]	[22]	[23]
5.4.1 измерения эталонных значений свойств	5.4.1.1 загрузки эталонного образца	0.9	0.3	0.5
	5.4.1.2 извлечения значений свойств эталонного образца	1	0.8	0.5
5.4.2 измерения свойств исследуемого образца	5.4.2.1 загрузки исследуемого образца	1	0.7	0.7
	5.4.2.2 извлечения значений свойств исследуемого образца	1	0.6	0.6
5.4.3 сопоставления полученных результатов с эталонными значениями	5.4.3.1 сравнения полученных значений с эталонными	1	1	1
	5.4.3.2 отображения результатов сравнения	1	1	0.7
	5.4.3.3 интерпретации результатов	0.1	0.7	0.9
Итоговая оценка		6	5.1	4.9

Аналоги [28-30] позволяют полностью интерпретировать код для последующего анализа и форматирования. Наибольшие возможности автоматического форматирования кода предоставляет аналог [28] Visual Studio 2017. Из настроек форматирования реализована только настройка отступов в аналогах [28], [30] Visual Basic for Applications, позволяя задать количество пробелов в табуляции. По результатам оценок, приведённых в таблице 9, прототипом подсистемы стандартизации оформления кода выбран аналог [28].

Таблица 9
Аналоги подсистемы стандартизации оформления кода

Критерии сравнения		Оценки аналогов по источникам информации		
Модуль	Блок	[28]	[29]	[30]
5.5.1 парсинга кода	5.5.1.1 выделения единиц языка	1	0.9	0.9
	5.5.1.2 выделения структур языка	1	1	1
5.5.2 форматирования кода	5.5.2.1 форматирования регистра	0.7	1	1
	5.5.2.2 форматирования отступов	1	0.2	0.3
	5.5.2.3 форматирования семантических структур	1	0.1	0.1
5.5.3 настройки форматирования	5.5.3.1 настройки форматирования регистра	0	0	0
	5.5.3.2 настройки форматирования отступов	1	0	0.5
	5.5.3.3 настройки форматирования семантических структур	0	0	0
Итоговая оценка		5.7	3.2	3.8

Стандарты [31], [32] в сравнении, приведённом в таблице 10, объединены в один, так как разработаны одной корпорацией и не противоречат друг другу. Они получили низкую оценку за простоту, в том числе из-за машинного перевода текста с английского.

Дата публикации оценивается по следующему правилу:

1) стандарты, опубликованные в 2018 году, оцениваются 1
2) за каждый год оценка снижается на 0.2 (до 0)

Таблица 10

Аналоги стандартов оформления кода

№	Критерии сравнения	Оценки аналогов по источникам информации				
		[31],[32]	[33]	[34]	[35]	[36]
5.5.1	Дата публикации	0.6	0	0.4	0.8	0
5.5.2	Подробность	1	0.8	1	0.1	1
5.5.3	Простота	0.5	1	0.5	1	1
5.5.4	Удобство	0.6	0.9	0.5	0.7	1
Итоговая оценка:		2.7	2.7	2.4	2.6	3

По итогам сравнения принято решение оформлять код корпоративного прототипа генератора на основе стандарта [36], при необходимости изменяя и дополняя его (в том числе другими стандартами [31-32]).

Для развития АГ СО ТЗ необходимо не только разработать систему шаблонов, но и интегрировать её с существующими системами. Для этого важно рассмотреть интеграцию аналогичных систем в других генераторах ТЗ [10-12] и [14]. В программе [10] – в качестве шаблона предлагается использовать заготовку – Sample.adl либо любой ранее созданный проект. Система шаблонов реализована как часть файловой системы; [11] – доступен только 1 неизменяемый шаблон. Сервис разработан «вокруг» этого шаблона, автоматизируя процесс его заполнения; [12] – содержит несколько шаблонов и позволяет импортировать документы, созданные в другом распространённых ПО; [14] – аналогично [11] позволяет заполнять единственный шаблон, который потом может быть использован как заготовка для ТЗ. Шаблон реализован в виде компонентов ЭФ.

Пакет прототипов

На рисунке 2 приведена иерархическая модель системы шаблонов. Вершины: 5 – система шаблонов, 5.1-4.5 подсистемы: 5.1 – содержания шаблона, 5.2 – каталогизации, 5.3 – формирования запроса к базе данных, 5.4 – оценки, 5.5 – стандартизации оформления кода; модули: 5.1.1 – разделение на ЭФ, 5.1.2 – размещения компонентов, 5.1.3 – размещения текста, 5.2.1 – ведения каталогов, 5.2.2 – обзора документов, 5.2.3 – взаимодействия с репозитарием, 5.3.1 – создания таблиц, 5.3.2 – создания полей, 5.3.3 – создания связей и ограничений, 5.4.1 – измерения эталонных значений свойств, 5.4.2 – измерения свойств исследуемого образца, 5.4.3 – сопоставления полученных результатов

с эталонными значениями, 5.5.1 – парсинга кода, 5.5.2 – форматирования кода, 5.5.3 – настройки форматирования.

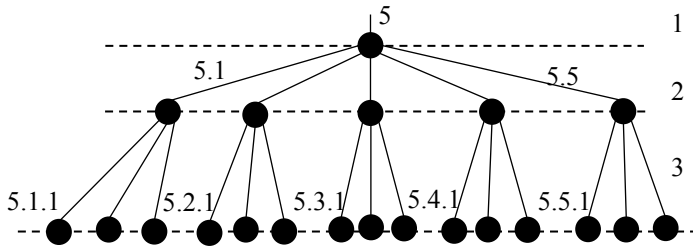


Рис. 2 Иерархическая модель системы шаблонов

В таблице 11 в колонке «название прототипа» при наличии нескольких указан прототип, набравший наибольшее количество баллов.

Таблица 11
Пакет прототипов

Ранг	Наименование	Источник	Название прототипа	Критика
0	Алгоритмическая модель генератора	[5]	Алгоритм функционирования АГ СО ТЗ	Нет блоков на обратной связи, не детализирована работа пользователя над ТЗ и его взаимодействие с генератором, не обладает требуемой гибкостью
1	Стандарт оформления кода	[36]	Стандарты и правила оформления кода C#	Недостаточно структурирован, избыточно регламентируется редко используемые конструкции
1	5 система шаблонов	[10]	Author IT	Нет функции модернизации экранных форм
2	5.3 подсистема формирования запроса к базе данных	[10], [28], [29]	Author IT	Работает только с интегрированной БД
2	5.2 подсистема каталогизации	[15], [16]	Where Is It?	Подсистемы не настроена на работу с шаблонами ТЗ на МИС
2	5.4 подсистема оценки качества	[21]	WinMerge 2.14.0	
2	5.5 подсистема стандартизации оформления кода	[28]	Visual Studio 2017	Нет функции загрузки пользовательских стандартов
3	5.2.3	[18-20]	GitHub	Не интегрирован с репозитарием и подсистемой каталогизации системы шаблонов
3	5.1.1 модуль разделения на ЭФ	[10], [12]	Help+Manual	Не поддерживают работу с компонентами
3	5.1.2 модуль размещения компонентов	[24], [25], [26]	Google Формы	Нет возможности произвольного размещения компонентов на форме
3	5.1.3 модуль размещения текста	[27], [12]	Microsoft Word 2016	Модуль не настроен на работу с ЭФ шаблона ТЗ на МИС

Результаты:

1. Модернизирован алгоритм работы АГ СО ТЗ

2. В алгоритм добавлены блоки анализа ситуации и анализа проблем
3. Предложено внести изменения в блок итоги
4. Выполнен анализ аналогов системы шаблонов и стандартов кода
5. Составлен пакет прототипов на систему шаблонов
6. Выбран стандарт кода

Вывод

Модернизированный алгоритм, система шаблонов и предложенный стандарт кода позволяют преодолеть выявленные проблемы программной реализации АГ СО ТЗ.

.

Список литературы

1. Покупка лицензионной версии AuthorIT / AuthorIT автоматизация документирования и разработки HTML-контента веб-сайтов. – 2014. URL:: http://authorit.ru/6948.

2. Методические рекомендации по обеспечению функциональных возможностей медицинских информационных систем медицинских организаций (МИС МО). – М.: Министерство здравоохранения Российской Федерации, 2016. – 82 с.

3. Автоматизированный генератор технического задания для врача-проектировщика : пат. 2465646 Рос. Федерация. № 2011107371/08, С.Л. Гольдштейн, Е.М. Грицюк, Д.А. Леонов; заявл. 25.02.11; опубл. 27.10.12, Бюл. № 30. 8 с. URL: http://www1.fips.ru/fips_servl/fips_servlet

4. Фирстков А.Л. Развитие системы шаблонов автоматизированного генератора системно обоснованного технического задания на информационные системы / А.Л. Фирстков, С.Л. Гольдштейн, Е.М. Грицюк // Электронный научный журнал «Системная интеграция в здравоохранении». 2017. №6. с.31-39.

5. Гольдштейн С.Л. О функционировании генератора системно обоснованного технического задания на медицинскую информационную систему / С.Л. Гольдштейн, Е.М. Грицюк, Д.А. Леонов // Электронный научный журнал «Системная интеграция в здравоохранении». 2012. №1. с. 20-32 URL: http://sys-int.ru/sites/default/files/sys_int_1_15_2012_0.pdf.

6. Гольдштейн С. Л. Автоматизированный генератор технического задания на медицинскую информационную систему / С. Л. Гольдштейн, Е. М. Грицюк, Г. В. Кожарская, Д. А. Леонов // Электронный научный журнал «Системная интеграция в здравоохранении». – 2011. – №1(11). – С. 43-57.

7. Пырин А.В., Предпосылки развития системы шаблонов в АГ СО ТЗ / А.В. Пырин, Е.М. Грицюк, С.Л. Гольдштейн // Тезисы доклада №1 IV ММНК «Физика. Технологии. Инновации», - Екатеринбург: УрФУ, 2017, – С. 14-15.

8. Свидетельство о государственной регистрации программы для ЭВМ: «Подсистема электронизации входной информации в автоматизированном генераторе системно-обоснованного технического задания» № 2013612950 от 19.03.13 г., Гольдштейн С.Л., Грицюк Е.М., Леонов Д.А.

9. Свидетельство о государственной регистрации программы для ЭВМ «Подсистема структуризации вербальных текстов в автоматизированном генераторе системно-обоснованного технического задания» №2018616223 от 02.06.2017, Гольдштейн СЛ., Грицюк Е.М., Бызова А.К.

10. Author IT / Author-it Software Corporation. – 2017. URL: http://www.author-it.com.

11. TZGen / Сервис автоматического создания технического задания на разработку веб-сайта. – 2018. URL: http://tzgen.ru/.

12. Help+Manual / Официальный сайт EC Software GmbH. – 2018. URL: https://www.helpandmanual.com.

13. Requirements Miner for AllFusion Modelling Suite / Interfaceru. Internet Software Company. 2008 URL: http://www.interface.ru/home.asp?artId=8811.

14. Мастер ТЗ / SoftPortal. – 2018. URL: http://www.softportal.com/software-5460-master-tehnicheskih-zadanij.html.

15. Where Is It? / Where Is It? Soft.com. – 2014. URL: http://www.whereisit-soft.com.

16. WinCatalog 2017 – универсальная программа-каталогизатор дисков, файлов и любых нефайловых элементов для Windows / WinCatalog.com. – 2017. URL: http://www.wincatalog.com/ru/.

17. Unicat editor (Universal Catalogizer) / Официальный сайт Unicat Editor. – 2004. URL: http://unicat.chat.ru.

18. Google Диск / Официальный сайт Google Drive. – 2018. URL: https://www.google.ru/drive/.

19. Dropbox / Официальный сайт Dropbox. – 2018. URL: https://www.dropbox.com.

20. GitHub / Официальный сайт GitHub. – 2018. URL: https://github.com/github.

21. WinMerge 2.14.0 / Официальный сайт WinMerge. – 2017. URL: http://winmerge.org/?lang=ru.

22. Istio. Сервис анализа текстов / Istio. – 2018. URL: https://istio.com/.

23. TopWriter. Сервис сравнения схожести текстов / Topwriter. – 2018. URL: http://www.topwriter.ru/comparison.

24. Google Формы: бесплатное создание собственных онлайн-опросов / Официальный сайт Google Forms. – 2017. URL: http://www.google.ru/forms/about/.

25. Online Form Builder JotForm / JotForm Inc. – 2017. URL: https://eu.jotform.com.

26. Inqwise. Free Survey Tool / Inqwise Online Survey Software. – 2017. URL: https://www.inqwise.com/en-us.

27. Microsoft Word 2016 / Официальный сайт Microsoft. – 2017. URL: http://www.microsoft.com.

28. Visual Studio 2017 / Официальный сайт Microsoft. – 2018. URL: https://www.visualstudio.com/ru/.

29. Microsoft Access / Официальный сайт Microsoft. – 2018. URL: https://products.office.com/ru-ru/access.

30. Visual Basic for Applications / Официальный сайт Microsoft. – 2018. URL: https://msdn.microsoft.com/ru-ru/library/office/ee814735(v=office.14).aspx.

31. Соглашения о написании кода на C# (Руководство по программированию на C#) / Официальный сайт Microsoft. URL: https://docs.microsoft.com/ru-ru/dotnet/csharp/programming-guide/inside-a-program/coding-conventions 20.07.2015

32. Правила именования / Официальный сайт Microsoft. URL : https://docs.microsoft.com/ru-ru/dotnet/standard/design-guidelines/naming-guidelines 30.03.2017.

33. C#: требования и рекомендации по написанию кода /сайт Хабр. – 2008. URL: https://habrahabr.ru/post/26077/.

34. Рекомендации по написанию кода на C# от Aviva Solutions / сайт Хабр. – 2015. URL: https://habrahabr.ru/post/272053/.

35. Правила оформления кода C# /сайт GitHub. – 2017. URL: https://github.com/gjrfytn/pk/wiki/Правила-оформления-кода-C%23.

36. Смирнов М. Стандарты и правила оформления кода C# / М. Смирнов //сайт msmirnov. – 2018. URL: http://www.msmirnov.ru/public/TSQL_Coding_Standards.doc

Фирстков Артем Леонидович - студент-бакалавр, гр. Фт-440007 ФГАОУ ВО «УрФУ имени первого Президента России Б.Н.Ельцина», г.Екатеринбург, ул.Мира, 19, тел. 8 912-264-38-79, e-mail:firstk121@gmail.com