

Министерство образования и науки Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Институт кибернетики
Направление подготовки информационные системы и технологии
Кафедра информационных систем и технологий

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Тема работы
Программное средство для автоматизированного заполнения баз данных тестовыми данными

УДК 004.415:004.6.054

Студент

Группа	ФИО	Подпись	Дата
8ИМ5А	Мочилин Кирилл Алексеевич		

Руководитель

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Мирошниченко Е.А	к.т.н		

КОНСУЛЬТАНТЫ:

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Попова С.Н.	к.э.н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент	Акулов П.А.			

ДОПУСТИТЬ К ЗАЩИТЕ:

Зав. кафедрой	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Мальчуков А.Н.	к.т.н.		

Томск – 2017 г.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ООП

Код результата	Результат обучения (выпускник должен быть готов)
Общепрофессиональные компетенции	
Р1	Воспринимать и самостоятельно приобретать, развивать и применять математические, естественнонаучные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте.
Р2	Владеть и применять методы и средства получения, хранения, переработки и трансляции информации посредством современных компьютерных технологий, в том числе в глобальных компьютерных сетях.
Р3	Демонстрировать культуру мышления, способность выстраивать логику рассуждений и высказываний, основанных на интерпретации данных, интегрированных из разных областей науки и техники, выносить суждения на основании неполных данных, анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями.
Р4	Анализировать и оценивать уровни своих компетенций в сочетании со способностью и готовностью к саморегулированию дальнейшего образования и профессиональной мобильности. Владеть, по крайней мере, одним из иностранных языков на уровне социального и профессионального общения, применять специальную лексику и профессиональную терминологию языка.
Профессиональные компетенции	
Р5	Разрабатывать стратегии и цели проектирования, критерии эффективности и ограничения применимости, новые методы, средства и технологии проектирования геоинформационных систем (ГИС) или промышленного программного обеспечения.
Р6	Планировать и проводить теоретические и экспериментальные исследования в области создания интеллектуальных ГИС и ГИС технологии или промышленного программного обеспечения с использованием методов системной инженерии.
Р7	Осуществлять авторское сопровождение процессов проектирования, внедрения и сопровождения ГИС и ГИС технологий или промышленного программного обеспечения с использованием методов и средств системной инженерии, осуществлять подготовку и обучение персонала.
Р8	Формировать новые конкурентоспособные идеи в области теории и практики ГИС и ГИС технологий или системной инженерии программного обеспечения. Разрабатывать методы решения нестандартных задач и новые методы решения традиционных задач. Организовывать взаимодействие коллективов, принимать управленческие решения, находить компромисс между различными требованиями как при долгосрочном, так и при краткосрочном планировании.
Общекультурные компетенции	
Р9	Использовать на практике умения и навыки в организации исследовательских, проектных работ и профессиональной эксплуатации современного оборудования и приборов, в управлении коллективом.
Р10	Свободно пользоваться русским и иностранным языками как средством делового общения.

P11	Совершенствовать и развивать свой интеллектуальный и общекультурный уровень. Проявлять инициативу, в том числе в ситуациях риска, брать на себя всю полноту ответственности.
P12	Демонстрировать способность к самостоятельному обучению новым методам исследования, к изменению научного и научно-производственного профиля своей профессиональной деятельности, способность самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности, способность к педагогической деятельности.

Министерство образования и науки Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Институт кибернетики

Направление подготовки (специальность) 09.04.02 «Информационные системы и технологии»

Кафедра Информационных систем и технологий

УТВЕРЖДАЮ:

Зав. кафедрой

(Подпись) _____

Мальчуков А.Н.
(Ф.И.О.)

ЗАДАНИЕ

на выполнение выпускной квалификационной работы

В форме:

магистерской диссертации

Студенту:

Группа	ФИО
8ИМ5А	Мочилину Кириллу Алексеевичу

Тема работы:

Программное средство для автоматизированного заполнения баз данных тестовыми данными

Утверждена приказом директора (дата, номер)

№ 897/с от 20.02.2017

Срок сдачи студентом выполненной работы:

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

Исходные данные к работе	Существующие системы для генерации тестовых данных
Перечень подлежащих исследованию, проектированию и разработке вопросов	Обзор и анализ аналогов, разработка требований к проектируемой системе, проектирование пользовательского интерфейса, проектирование модулей для генерации данных, проектирование служебной базы данных, проектирование модуля автоматизированного подбора стартовых параметров, проектирование логики для поддержки ограничений целостности, реализация функционала для генерации данных для представлений, проектирование модуля

	настройки пользовательских генераторов данных.
Перечень графического материала	
Консультанты по разделам выпускной квалификационной работы	
Раздел	Консультант
Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	Попова С.Н.
Социальная ответственность	Акулов П.А.
Раздел на иностранном языке	Горбатова Т.Н.
Названия разделов, которые должны быть написаны на русском и иностранном языках:	
Разделы введение, анализ предметной области, описание программной реализации, финансовый менеджмент, ресурсоэффективность и ресурсосбережение, социальная ответственность и заключение должны быть написаны на русском языке.	
Часть раздела описание программной реализации должна быть написана на английском языке.	

Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	
---	--

Задание выдал руководитель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент каф. ИСТ	Мирошниченко Е.А.	к.т.н.		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ИМ5А	Мочилин К.А.		

Министерство образования и науки Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Институт кибернетики

Направление подготовки (специальность) 09.04.02 «Информационные системы и технологии»

Уровень образования: магистратура

Кафедра Информационных систем и технологий

Период выполнения: осенний семестр 2016 г. – весенний семестр 2017 учебного года

Форма представления работы:

магистерская диссертация

**КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН
выполнения выпускной квалификационной работы**

Срок сдачи студентом выполненной работы:

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
23.06.2016	<i>Постановка задачи и анализ предметной области</i>	10
30.08.2016	<i>Формирование требований</i>	10
25.09.2016	<i>Проектирование архитектуры</i>	15
11.11.2016	<i>Проектирование служебной базы данных</i>	15
14.03.2017	<i>Проектирование модулей генерации данных</i>	20
15.05.2017	<i>Проектирование пользовательского интерфейса</i>	20
01.06.2017	<i>Оформление пояснительной записки</i>	5

Составил преподаватель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент каф. ИСТ	Мирошниченко Е.А.	к.т.н.		

СОГЛАСОВАНО:

Зав. кафедрой	ФИО	Ученая степень, звание	Подпись	Дата
ИСТ	Мальчуков А.Н.	к.т.н.		

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСО-
СБЕРЕЖЕНИЕ»**

Студенту:

Группа	ФИО
8ИМ5А	Мочилин Кирилл Алексеевич

Институт	Кибернетики	Кафедра	Информационных систем и технологий
Уровень образования	Магистратура	Направление/специальность	09.04.02 Информационные системы и технологии

Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:

1. Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих
2. Нормы и нормативы расходования ресурсов
3. Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования

Перечень вопросов, подлежащих исследованию, проектированию и разработке:

Оценка коммерческого и инновационного потенциала НТИ	Оценка конкурентоспособности разработки, анализ перспективности проекта, оценка готовности проекта к коммерциализации.
Организация и планирование работ	Планирование этапов разработки программы, построение календарного плана, построение диаграммы Ганта.
Формирование бюджета научных исследований	Расчёт сметы затрат на выполнение проекта.
Определение ресурсной, финансовой, экономической эффективности	Оценка научно-технического уровня проекта.

Дата выдачи задания для раздела по линейному графику	
--	--

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент каф. МЕН	Попова Светлана Николаевна	к.э.н.		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ИМ5А	Мочилин Кирилл Алексеевич		

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА
«СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»**

Студенту:

Группа	ФИО
8ИМ5А	Мочилин Кирилл Алексеевич

Институт	Кибернетики	Кафедра	Информационных систем и технологий
Уровень образования	Магистратура	Направление/специальность	Информационные системы и технологии

<ul style="list-style-type: none"> Исходные данные к разделу «Социальная ответственность»: 	
<p>1. Характеристика объекта исследования и области его применения</p>	<p>Программное средство для автоматизированного заполнения базы данных тестовыми данными. Область применения: автоматизация тестирования при разработке программного обеспечения, баз данных.</p>
<ul style="list-style-type: none"> Перечень вопросов, подлежащих исследованию, проектированию и разработке: 	
<p>1. Профессиональная и социальная безопасность</p> <p>1.1 Анализ вредных и опасных факторов, которые могут возникнуть на рабочем месте</p> <p>1.2 Обоснование мероприятий по защите от действия опасных и вредных факторов</p>	<p>1.1 Вредные факторы при работе с компьютером:</p> <ul style="list-style-type: none"> недостаток естественного света; недостаточная освещенность рабочей зоны монотонность труда. <p>Опасные факторы при работе с компьютером:</p> <ul style="list-style-type: none"> поражение электрическим током; пожар. <p>1.2 Рекомендации по обеспечению безопасности при работе с компьютером</p>
<p>2. Экологическая безопасность</p> <p>2.1 Анализ влияния объекта исследования на окружающую среду</p> <p>2.2 Обоснование мероприятий по защите окружающей среды</p>	<p>2.1 Предполагаемые источники загрязнения: устаревшая и/или пришедшая в негодность компьютерная техника.</p> <p>2.2 Мероприятия по утилизации отходов комплекующих.</p>
<p>3. Безопасность в чрезвычайных ситуациях</p> <p>3.1 Анализ вероятных ЧС, которые могут возникнуть на рабочем месте</p>	<p>3.1 Техногенные пожары, природные пожары.</p>

3.2 Обоснование мероприятий по предотвращению ЧС и разработка порядка действия в случае возникновения ЧС	3.2 Мероприятия по предотвращению и порядок действий при возникновении пожара.
4. Правовые и организационные вопросы обеспечения безопасности 4.1 Специальные правовые нормы трудового законодательства 4.2 Влияние разработки на организацию производственных процессов	4.1 Рабочее место при выполнении работ в положении сидя должно соответствовать требованиям проведения работ на ПЭВМ. 4.2 Плюсы от внедрения разработки с точки зрения сокращения монотонного труда.

Дата выдачи задания для раздела по линейному графику	
--	--

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент	Акулов П.А.			

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8ИМ5А	Мочилин К.А.		

Реферат

Выпускная квалификационная работа содержит 117 страниц, 40 рисунков, 22 таблицы, 25 источников, 1 приложение.

Ключевые слова: информационные системы (ИС), базы данных (БД), генерация данных, автоматизация тестирования.

Объектом исследования является процесс программной генерации тестовых данных для БД.

Цель работы – проектирование генератора тестовых данных для Microsoft SQL Server.

В процессе исследования были изучены и проанализированы существующие программные аналоги для генерации тестовых данных.

В результате исследования было спроектировано и реализовано программное средство для автоматизированного заполнения баз данных тестовыми данными.

Область применения: автоматизация тестирования ИС.

СОДЕРЖАНИЕ

Введение.....	15
1. Анализ предметной области	17
1.1 Рассмотрение аналогов.....	18
1.1.1 Toad for SQL Server freeware 6.6.....	19
1.1.2 EMS Database Management Solutions: Data Generator for SQL Server	20
1.1.3 Sybase PowerDesigner.....	22
1.1.4 Redgate SQL Data Generator.....	24
1.1.5 dbForge Studio for SQL Server	25
1.1.6 Веб-сервис generatedata.com	26
1.1.7 Сравнительный анализ программных решений.....	28
1.2 Требования к разрабатываемой системе.....	30
1.3 Генерация данных для представлений.....	32
1.4 Ограничения целостности данных	34
1.5 Автоматизированный подбор стартовых параметров.....	36
2. Описание программной реализации.....	37
2.1 Декомпозиция задачи.....	37
2.2 Архитектура программной системы	37
2.3 Пользовательский интерфейс	40
2.4 Примеры работы программы	41
2.5 Основные алгоритмы работы.....	47
2.6 Служебная база данных.....	52
2.7 Соблюдение ограничений целостности целевой БД.....	54
2.8 Модуль автоматизированного подбора стартовых параметров.....	56

2.9 Проектирование интерфейса пользователя для модулей генерации данных	60
2.10 Генерация данных для представлений.....	64
2.11 Модуль задания пользовательских типов генераторов.....	67
2.12 Пример работы модуля пользовательских генераторов данных	68
3. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение..	71
3.1. Оценка коммерческого и инновационного потенциала НТИ.....	71
3.1.1 Потенциальные потребители	71
3.1.2 Анализ конкурентных технических решений с позиции ресурсоэффективности и ресурсосбережения.....	72
3.1.3 FAST-анализ	73
3.1.4 Оценка готовности проекта к коммерциализации.....	78
3.2 Организация и планирование работ	79
3.3 Расчёт сметы затрат на выполнение проекта	80
3.3.1 Расчёт заработной платы.....	80
3.3.2 Материальные затраты	82
3.3.3 Отчисления на социальные нужды	82
3.3.4 Затраты на электроэнергию.....	83
3.3.5 Накладные расходы.....	84
3.3.6 Расчет общей себестоимости разработки	84
3.4 Оценка научно-технического уровня проекта	84
4. Социальная ответственность	86
4.1. Профессиональная социальная безопасность	87
4.1.1 Анализ вредных и опасных факторов на рабочем месте	87
4.1.1.1 Вредные производственные факторы	88

4.1.1.1.1 Недостаток естественного света.....	88
4.1.1.1.2 Недостаточная освещенность рабочей зоны.....	89
4.1.1.1.3 Монотонность труда.....	90
4.1.1.2 Опасные производственные факторы.....	91
4.1.1.2.1 Опасность поражения электрическим током.....	91
4.1.1.2.2 Опасность возникновения пожара.....	91
4.1.2 Обоснование мероприятий по защите от действия вредных и опасных факторов.....	92
4.1.2.1 Мероприятия по улучшению условий труда по фактору световая среда.....	92
4.1.2.2 Мероприятия по улучшению условий работы в монотонном режиме.....	93
4.1.2.3 Мероприятия по предотвращению возможности поражения электрическим током.....	94
4.1.2.4 Мероприятия по профилактике пожаров.....	95
4.2. Экологическая безопасность.....	95
4.2.1 Анализ влияния объекта исследования на окружающую среду.....	95
4.2.2 Обоснование мероприятий по защите окружающей среды.....	96
4.3 Безопасность в чрезвычайных ситуациях.....	97
4.3.1 Анализ вероятных ЧС, которые могут возникнуть на рабочем месте ...	97
4.3.2 Обоснование мероприятий по предотвращению ЧС и разработка порядка действия в случае возникновения ЧС.....	98
4.4 Правовые и организационные вопросы обеспечения безопасности.....	99
4.4.1 Специальные правовые нормы трудового законодательства.....	99
4.4.2 Влияние разработки на организацию производственных процессов...	101
Заключение.....	102

Список использованных источников	103
Приложение А	107

ВВЕДЕНИЕ

Задача генерации тестовых данных актуальна для всех разработчиков информационных систем (ИС). При разработке и последующем тестировании информационной системы возникает потребность наполнения баз данных (БД) системы тестовыми данными. Тестовые данные в БД необходимы для тестирования создаваемых ИС и для демонстрации их возможностей.

Цель работы – разработка генератора тестовых данных для Microsoft SQL Server.

Для достижения поставленной цели необходимо решить следующие задачи:

- провести анализ предметной области на предмет существующих аналогов, выявить их сильные и слабые стороны;
- разработать требования к проектируемой системе;
- спроектировать пользовательский интерфейс программы;
- реализовать модули для генерации данных различных типов;
- реализовать модуль для автоматизированного подбора стартовых параметров (АПСП) системы;
- внедрить в разрабатываемую систему служебную базу данных для хранения системных данных и пользовательских настроек;
- добавить в разрабатываемую систему логику для поддержки ограничений целостности целевой БД;
- реализовать функционал для генерации данных для представлений;
- реализовать модуль для настройки пользовательских генераторов данных.

Для обеспечения повышения производительности в терминах автоматизации производства данных система должна обладать пользовательским интерфейсом, сочетающим в себе одновременно удобство использования и возможность гибкой настройки параметров. Разрабатываемая система также должна отвечать определенным требованиям бизнес-логики, к которым, в частности,

относятся: релевантность генерируемых данных, соблюдение целостности данных.

Описание задачи генерации данных и некоторых её особенностей, а также сравнение аналогов содержится в разделе 1.

Раздел 2 посвящена описанию разрабатываемой системы, основных алгоритмов, и примеров работы системы.

В разделах 3 и 4 приведены материалы, связанные с ресурсоэффективностью и социальной ответственностью.

1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

Задача генерации тестовых данных возникает перед разработчиками и тестерами программного обеспечения (ПО), в том числе и при разработке баз данных.

Задача тестирования очень тесно связана с тестовыми данными, которые требуется генерировать, либо писать самостоятельно. Особенно актуальной задача генерации становится при нагрузочном тестировании, когда необходимо внести в систему не несколько штук или десятков, а миллионы объектов. Многие разработчики, сталкиваясь с задачей генерации тестовых данных занимаются ручным написанием скриптов для генерации данных, однако этот способ не является идеальным, поскольку он не универсален, и написанный в результате такого подхода исходный код трудно сопровождать [1].

Существует отдельный класс программных средств, предназначенных для решения задачи генерации тестовых данных. Применяемые этими программами способы генерации и заполнения БД тестовыми данными можно свести к следующему:

- Выборка значений из заранее заданного множества.
- Генерация псевдослучайных чисел в реальном времени.
- Выборка случайных значений из SQL-запроса.
- Выборка случайных значений по внешнему ключу.
- Инкрементирование значений.
- Различные комбинации вышеупомянутых способов.

При генерации тестовых данных для БД, часто возникает вопрос о релевантности генерируемых данных. Под «релевантностью» подразумевается смысловое соответствие между информационным запросом и полученным сообщением, а также «способность информации соответствовать потребностям пользователя» [2]. В контексте генерации тестовых данных для БД, являются релевантными те данные, которые не нарушают ограничений целостности целевой БД, и в то же время являются, насколько это возможно, приближенными

к реальным данным, то есть как минимум тип генерируемых данных должен совпадать с типом данных, заложенным разработчиком при проектировании БД. Кроме того, для данных, относящихся к определенному атрибуту сущности, должна быть возможность генерации значений, близких по смыслу к реальным «рабочим» значениям. Как пример, для атрибута «Имя» программа должна предлагать набор реальных имён, а не просто случайный набор символов.

Задача обеспечения релевантности генерируемых данных относится к реализации изначальной цели разрабатываемой системы – облегчению труда разработчика баз данных, и, как следствие, повышению качества разрабатываемых систем. Релевантные данные более упорядоченно воспринимаются визуально, чем хаотичные и случайные, что позволяет быстрее и лучше замечать некоторые виды ошибок в ходе тестирования. Кроме того, такие данные предпочтительнее для демонстрации возможности разрабатываемой ИС.

Кроме обеспечения релевантности данных, существует ряд других задач, связанных с генерацией данных для ИС, эти задачи будут рассмотрены позднее.

1.1 Рассмотрение аналогов

Для формирования требований к программной системе были рассмотрены следующие программные решения для генерации тестовых данных:

- 1) Toad for SQL Server freeware 6.6
- 2) EMS Database Management Solutions: Data Generator for SQL Server
- 3) Sybase PowerDesigner
- 4) RedGate SQL Data Generator – входит в состав Red Gate SQL Toolbelt
- 5) dbForge Studio for SQL Server
- 6) Веб-сервис generatedata.com

1.1.1 Toad for SQL Server freeware 6.6

Toad — программное обеспечение, разрабатываемое корпорацией Dell (перешедшее от поглощённой компании Quest), используемое для работы с реляционными системами управления базами данных. Располагает редактором исходного текста и данных, наборами данных для изменения таблиц и обозревателем для просмотра и управления объектами базы данных в графическом режиме [5].

В данной главе рассматривается инструмент DataGenerate программного комплекса Toad for SQL Server.

Помимо генерации данных путем выбора случайных значений из тематических словарей, поддерживаются регулярные выражения (RegularExpression), возможность использовать результаты запроса SQL (SQL-Expression), выборка случайных значений из множества записей по внешнему ключу, импорт и экспорт данных в форматах CSV, Excel, SQL.

Есть возможность сохранять и загружать собственные шаблоны генерации данных.

Пример одного из вариантов оконного интерфейса для генерации данных в Toad представлен на рис. 1.

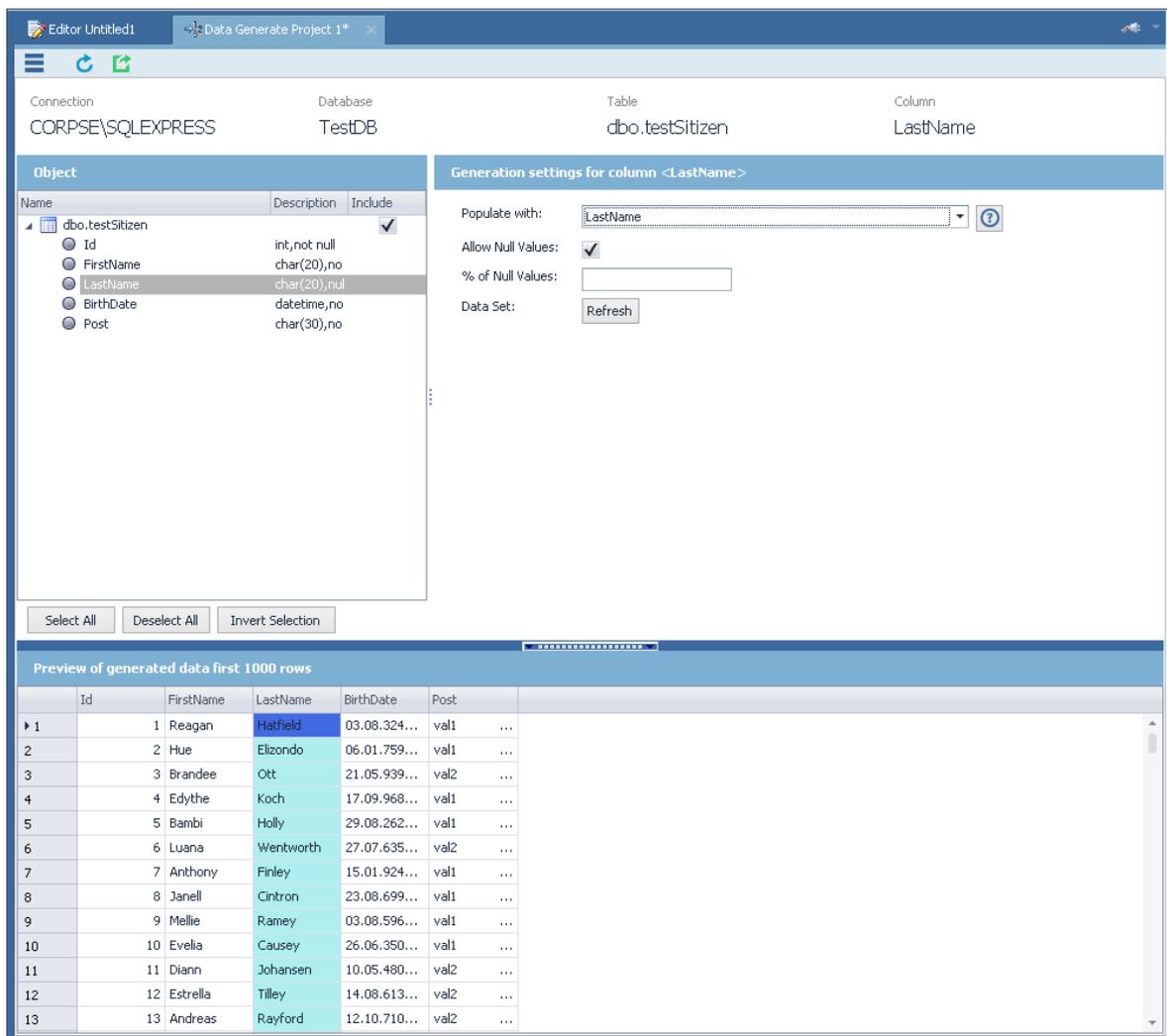


Рис. 1. Пример оконного интерфейса для генерации данных в Toad

В программе не предусмотрена возможность работы с UNIQUE-KEY полями, DataGenerate будет генерировать значения для этих полей, не учитывая требование уникальности.

1.1.2 EMS Database Management Solutions: Data Generator for SQL Server

При помощи приложения Data Generator for SQL Server (далее – Data Generator) можно моделировать среду рабочей базы данных, заполнять одновременно несколько таблиц SQL Server тестовыми данными, задавать таблицы для генерации данных, устанавливать диапазоны значений, создавать текстовые поля по маске, задавать списки значений вручную или выбирать их из запросов

SQL, настраивать различные параметры генерации тестовых данных SQL Server для каждого типа поля. Data Generator предоставляет инструменты для генерации тестовых данных в базах SQL Server и дополнительно включает консольную утилиту для генерации тестовых данных с помощью шаблонов генерации.

Есть возможность задать порядок генерации данных в таблицы. Таким образом можно соблюсти зависимости данных. Также, Data Generator анализирует порядок генерации данных, с целью предотвратить нарушение целостности ссылочных данных, и советует изменить порядок их генерации.

Есть возможность включить игнорирование триггеров в таблицах при генерации данных.

Для полей типа INTEGER предусмотрена возможность задавать формулы вида $value = f(x)$, где $value$ —желаемое значение, x — число, сгенерированное случайным образом. Можно использовать операторы сложения, вычитания, умножения, деления и возведения в степень (+, -, *, /, ^). [6]. Интерфейс программы приведен на рис. 2.

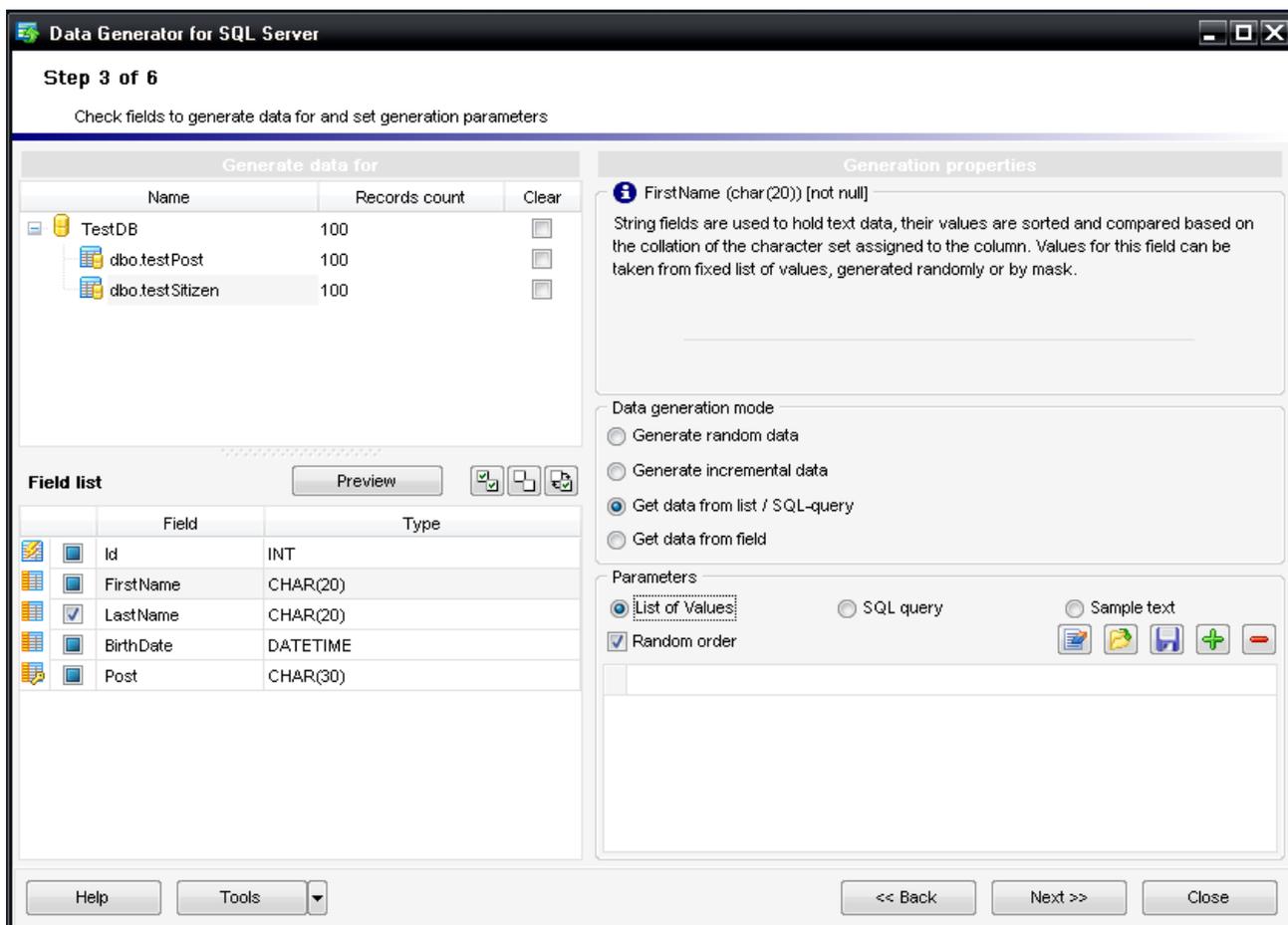


Рис. 2. Оконный интерфейс Data Generator for SQL Server

1.1.3 Sybase PowerDesigner

PowerDesigner – полнофункциональный инструмент для создания бизнес-приложений, включающий в себя средства моделирования бизнес-процессов, возможности концептуального и физического проектирования баз данных, возможности моделирования с использованием UML, и предоставляющий централизованный репозиторий для хранения моделей и объектов.

При создании диаграммы физической модели данных (Physical DataModel, PDM) Power Designer позволяет использовать встроенный инструмент генерации тестовых данных для БД. Оконный интерфейс данного инструмента показан на рис. 3.

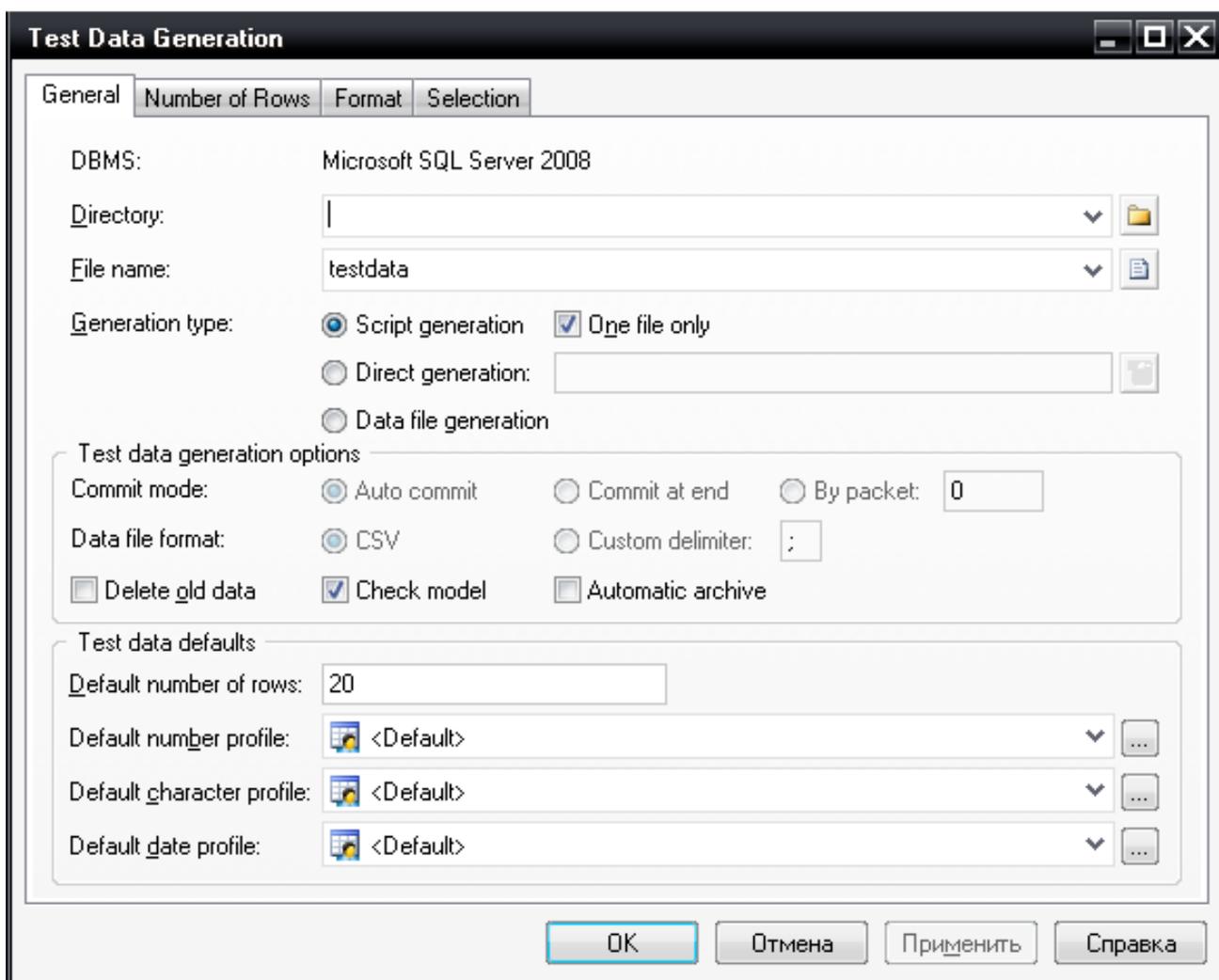


Рис. 3. Диалоговое окно для генерации тестовых данных для БД в Sybase PowerDesigner

PowerDesigner может генерировать данные для таблиц базы данных, чтобы проверить производительность или помочь в оценке объема памяти, который потребует база данных. Можно генерировать тестовые данные для некоторых или всех таблиц в PDM в пустую или существующую базу данных.

PowerDesigner поддерживает большое число СУБД, к недостаткам такой универсальности можно отнести относительно небогатую (в сравнении с другими программными продуктами в данном обзоре) вариативность генерируемых данных. В частности, на одной из страниц официального информационного центра Sybase приводится следующее замечание: «The following objects are not taken into account when you generate test data: Alternate keys; Foreign keys;

Business and validation rules, Binary, sequential, OLE, text or image data types; Trigger contents» [7].

1.1.4 Redgate SQL Data Generator

Программа, подобно своим аналогам, автоматически предлагает различные варианты для генерации, в зависимости от типа выбранного поля.

Есть возможность настройки генерации данных по внешнему ключу: пользователю предоставляется выбор между генерацией из внешнего поля связанной таблицы (automatic), внешнего поля другой (несвязанной) таблицы (manual), и генерацией случайной ASCII-строки.

SQL DataGenerator поддерживает скрипты на языке Python. На рис. 4 показан предлагаемый программой базовый шаблон скрипта для генерации случайных строковых данных.

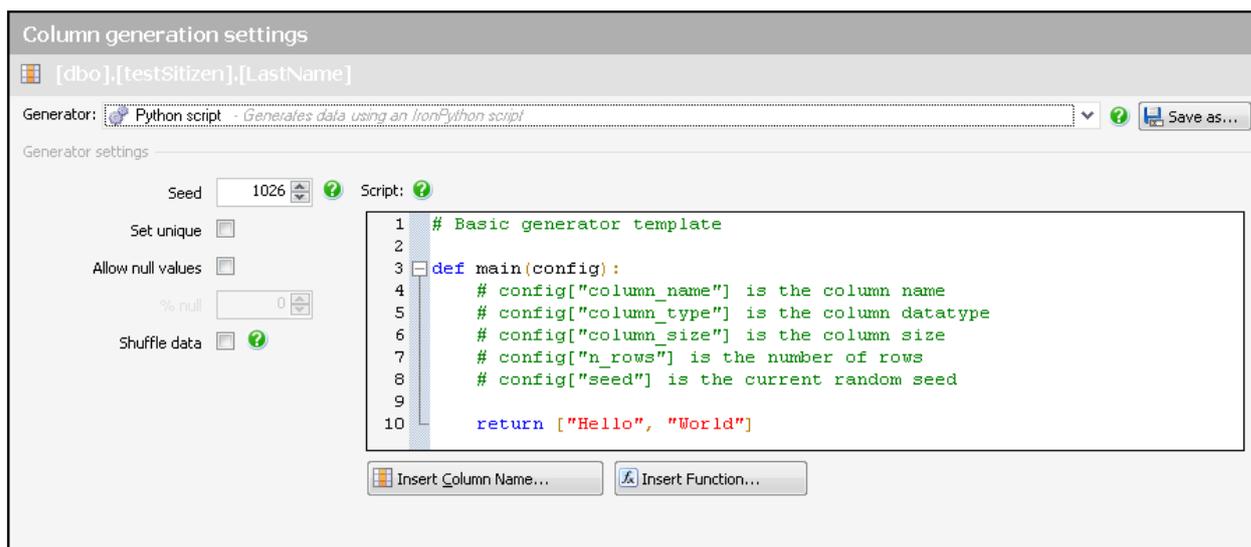


Рис. 4. Шаблон python-скрипта для генерации данных в Redgate SQL DataGenerator

Поддерживается импорт данных из cvs формата.

По окончании работы программа представляет пользователю отчёт, в котором содержится информация по каждой таблице о количестве сгенерированных строк, времени старта и окончания работы, причинах преждевременной остановки и т.п.

Цена – один из недостатков программы, производителем установлена суммарная стоимость продукта для одного пользователя с годом технической поддержки, равная 369 долл. США в случае standalone варианта, 2376 долл. США в составе пакета SQLToolBelt. Также возможно свободное использование в течение 14-дневного trial-периода [8].

1.1.5 dbForge Studio for SQL Server

Среда dbForge Studio for SQL Server – это полнофункциональная среда разработки для управления SQL Server, администрирования, создания отчетов и анализа данных. Данный инструмент ускоряет выполнение повседневных задач, позволяет вносить сложные изменения в базы данных, а также управлять изменениями баз данных в популярных системах контроля версий [9].

Встроенный в данную среду разработки инструмент генерации данных обладает следующими возможностями:

- Полная поддержка всевозможных типов данных.
- Создание нескольких генераторов с разными общими настройками.
- Автоматическая поддержка заданных ограничений целостности БД, таких как внешние ключи и уникальные значения. Однако, триггеры и проверочные ограничения (CHECK CONSTRAINT) не учитываются во время генерации, дабы избежать конфликтов с внутренней логикой БД.
- Скрипты генерации данных могут быть сохранены на диск в виде SQL-файла.
- Интерфейс для работы через командную строку.

Из особенностей данного инструмента для генерации данных следует отметить элементы интеллектуального анализа табличных метаданных, результатом которого является автоматически предложенные генераторы для полей, имеющих определенный тип и имя столбца. Например, для имени столбца «URL документ» программа автоматически предлагает генератор «URL (web)».

Графический интерфейс генератора данных dbForgeStudio показан на рис. 5.

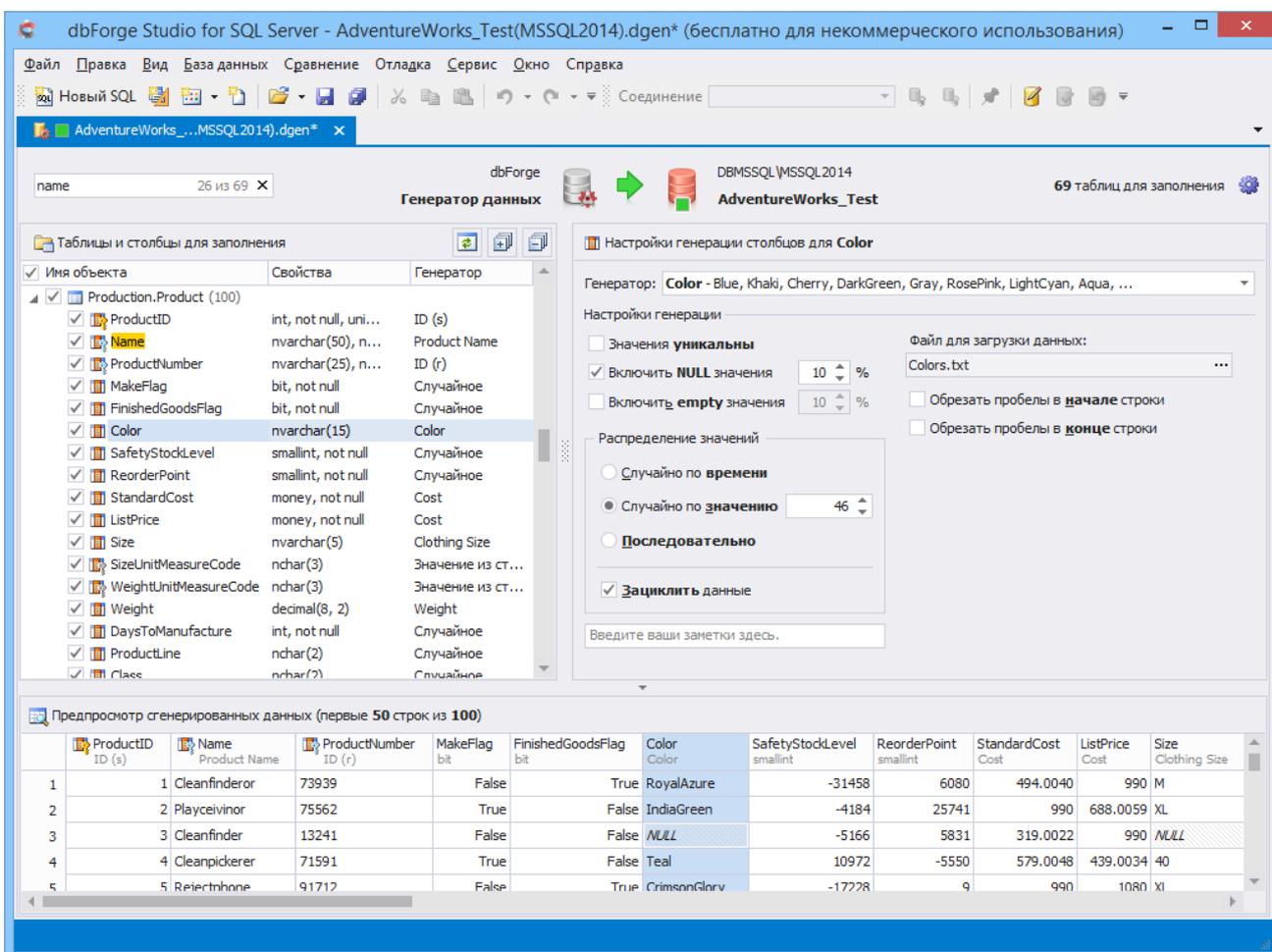


Рис. 5. Графический интерфейс генератора данных dbForgeStudio

1.1.6 Веб-сервис generatedata.com

Веб-сервис представляет собой набор скриптов для генерации данных в различных форматах под разные СУБД.

Поддерживаемые форматы экспорта данных:

CSV, Excel, HTML, JSON, LDIF, XML, JavaScript, Perl, PHP, Ruby, SQL (есть возможность выбора между MySQL, Postgres, SQLite, Oracle, SQL Server);

Предоставляемые сервисом возможности для генерации, в сравнении с коммерческими решениями, довольно скудны. В частности, за одну “сессию” генерации можно сгенерировать данные только для одной таблицы или одного

набора данных. Кроме того, для каждой такой генерации приходится настраивать все параметры вручную, что отнимает значительное время.

Предоставленная онлайн версия является пробной и имеет ограничение в 100 одновременно генерируемых строк, однако есть возможность бесплатно скачать скрипты и установить их на локальный веб-сервер для того чтобы снять это ограничение.

Графический интерфейс показан на рис. 6.

The screenshot shows the web interface of generatedata.com. At the top, there is a navigation bar with 'Generate', 'About', 'News', and 'Donate' buttons. Below this is a form for 'Your data set name here...' with a 'SAVE' button and icons for a calendar, a refresh, and a link. The 'COUNTRY-SPECIFIC DATA' section has a dropdown menu set to 'All countries'. The 'DATA SET' section contains a table with columns: Order, Table Column, Data Type, Examples, Options, Help, and Del.

Order	Table Column	Data Type	Examples	Options	Help	Del
1	Person	Names	Alex (any gender)	Name	?	□
2	Post	Custom List	Department Names	Exactly 2 At Most 1 Accounting Advertising Asset Management Custo	?	□
3	Salary	Number Range	No examples available.	Between 9000 and 15000	?	□
4	Id	GUID	No examples available.	No options available.	?	□

Below the table is an 'Add 1 Row(s)' button. The 'EXPORT TYPES' section has tabs for CSV, Excel, HTML, JSON, LDIF, Programming Language, SQL, and XML. Under the SQL tab, there are settings for 'Database table name' (myTable), 'Database Type' (SQL Server), 'Statement Type' (INSERT), 'INSERT batch size' (100), 'Primary Key' (None), and 'Misc Options' (checked: Include CREATE TABLE query, Include DROP TABLE query, Enclose table and field names with backquotes). At the bottom, there is a 'Generate' button and options for 'Generate 100 rows', 'Generate in-page', 'New window/tab', 'Prompt to download', and 'Zip?'. The footer contains the version '3.2.4' and links for 'Documentation' and 'Report a bug'.

Рис. 6. Веб-сервис generatedata.com

1.1.7 Сравнительный анализ программных решений

Был сформирован набор основных требований к программам для генерации тестовых данных, на основе этих требований были выделены следующие критерии для проведения сравнительного анализа:

- Словари – возможность делать выборку случайных текстовых данных из тематических словарей.
- Регулярные выражения – поддержка регулярных выражений (Regular Expression).
- Unique-поля – если поле помечено как UNIQUE-KEY, то генератор не должен предлагать повторяющиеся значения для данного атрибута.
- Внешние ключи – выборка случайных значений из множества записей по внешнему ключу.
- Зависимости – учет зависимостей между таблицами в БД. Генерация данных должна происходить сначала в таблицах, содержащих внешние ключи, затем – в зависимых от этих ключей таблицах.
- Скрипты – поддержка различных скриптовых языков программирования.
- Предпросмотр/редактирование – просмотр сгенерированных данных с возможностью их изменения до внесения в БД.
- Импорт/экспорт данных – сохранение данных в файл на диске и загрузка данных из файла.

В таблице 1 отмечено соответствие (+), несоответствие (–), либо частичное соответствие (+/–) рассмотренных в обзоре программных решений упомянутым требованиям.

Таблица 1. Сравнительный анализ существующих решений генерации тестовых данных для СУБД MS SQL Server

	Dell Toad for SQL Server	EMSDat-aGenerator	Sybase Power Designer	Redgate SQL Data Generator	dbForge Studio for SQL Server	Веб-сервис generatedata.com
Словари	+	+/-	-	+	+	+/-
Регулярные выражения	+	-	-	+	+	-
Unique-поля	-	-	-	+	+	-
Внешние ключи	+	+	-	+	+	-
Зависимости (ссылки)	-	+	-	+	+	-
Скрипты	-	-	-	+	+	+/-
Возможность предпросмотра и редактирования	+/-	+/-	-	+/-	+/-	+
Импорт данных	+	-	+	+	+	-
Экспорт данных	+	-	+	-	+	+

Из рассмотренных программных решений лишь несколько, а именно Redgate SQL Data Generator и dbForge Studio for SQL Server, наиболее полно отвечают выдвинутым требованиям. Однако, эти решения не лишены недостатков, в частности, нет возможности редактировать данные вручную в режиме предварительного просмотра.

Кроме того, ни одна рассмотренных из программ не отвечает следующим потребностям:

- настройка автоматизированного подбора стартовых параметров генерации;
- учёт ограничений Check Constraint;
- возможность генерации данных для View (представлений).

На основе анализа рассмотренных решений были сформированы требования к разрабатываемому программному продукту, в ходе реализации которого планируется избавиться от замеченных недостатков. Данные требования приводятся в главе 2.8.

1.2 Требования к разрабатываемой системе

Целевая система: программное решение, обеспечивающее генерацию тестовых данных для баз данных.

Цель системы: обеспечить удобный и функциональный инструментарий, позволяющий генерировать, редактировать и управлять тестовыми данными при тестировании и разработке базы данных под управлением СУБД MS SQL Server.

Требования разбиты по группам.

Группы требований:

1. К пользовательскому интерфейсу – требования к оконному графическому интерфейсу программы, выполнение которых призвано облегчить решение стоящих перед пользователем задач.
2. К функциональным возможностям системы
 - 2.1 Общие требования.
 - 2.2 Варианты генерации – настраиваемые возможности для обеспечения потребностей пользователя.
 - 2.3 Поддержка дополнительных возможностей – возможности, обеспечивающие ограничения целостности БД.

Требования к пользовательскому интерфейсу

1. Минимальное разрешение окон программы, при котором в окне помещаются все элементы, должно составлять 1280x1024 пикселя.
2. Окна графического интерфейса программы должны масштабироваться корректно.

Общие требования к функциональным возможностям

1. Программа должна обеспечить возможность работы в режиме проекта.
2. Программа должна обеспечить подключение к удаленному серверу БД с учетом параметров строки подключения.

3. Пользователь должен иметь возможность указать число генерируемых строк для каждой таблицы.
4. Программа должна иметь возможность настройки параметров в интеллектуальном автоматическом режиме.
5. По окончании генерации программа должна представить пользователю отчет о проделанной работе.

Требования к вариантам генерации

1. Для каждого поля таблицы программа должна предлагать перечень доступных вариантов генерации в зависимости от типа поля и заданных ограничений целостности.
2. Для текстовых полей программа должна поддерживать подстановку случайных значений из тематических словарей, поставляющихся в составе системы. Также должна быть предусмотрена возможность добавления и редактирования словарей пользователем.
3. Для числовых полей программа должна поддерживать функцию автоинкремента, задание случайного распределения по выбранному закону с заданными параметрами, задание значений на основе регулярного выражения и наполнение значениями заданной константы или значениями по умолчанию для столбца.

Требования к поддержке ограничений целостности БД

1. Программа должна поддерживать генерацию уникальных, неповторяющихся значений для полей с модификатором UNIQUE.
2. В том случае, если поле таблицы соединено внешним ключом с полем другой таблицы, программа должна помимо прочих вариантов предложить генерацию случайных значений из набора значений по внешнему ключу.
3. Генерация данных должна происходить сначала в таблицах, содержащих внешние ключи, затем – в зависимых таблицах.
4. Программой должны приниматься во внимание системные ограничения таблиц типа CHECK CONSTRAINT.

5. В программе должен быть предусмотрен режим генерации для View (представлений).

1.3 Генерация данных для представлений

В SQL существует два вида таблиц: базовые таблицы (base table) и представления (view). Представления – это таблицы чье содержание выбирается или получается из других таблиц. Они работают в запросах и операторах SQL точно также, как и основные таблицы, но не содержат никаких собственных данных. Представление – это фактически запрос, который выполняется всякий раз, когда представление становится темой команды. Вывод запроса при этом в каждый момент становится содержанием представления.

Представления могут быть модифицируемыми и немодифицируемыми [3]. Модифицируемость представления означает возможность выполнения любой из трех команд модификации («Вставить», «Изменить» и «Удалить»).

Представление может изменяться командами модификации, но модификация не будет воздействовать на само представление. Команды будут перенаправлены к базовой таблице.

Имеются типы представлений, которые являются доступными только для чтения. Это означает, что их можно запрашивать, но они не могут подвергаться действиям команд модификации.

Один из наиболее трудных и неоднозначных аспектов представлений - непосредственное их использование с командами модификации. Эти команды фактически воздействуют на значения в базовой таблице представления. Это является некоторым противоречием. Представление состоит из результатов запроса, и когда модифицируется представление, то модифицируется набор результатов запроса. Но модификация не должна воздействовать на запрос; она должна воздействовать на значения в таблице к которой был сделан запрос, и таким образом изменять вывод запроса.

Определение того, является ли представление модифицируемым, в теории баз данных пока обсуждаемая тема. Модифицируемое представление – это представление в котором команда модификации может выполняться, чтобы изменить одну и только одну строку основной таблицы в каждый момент времени, не воздействуя на любые другие строки любой таблицы. Использование этого принципа на практике, однако, затруднено. Кроме того, некоторые представления, которые являются модифицируемыми в теории, на самом деле не являются модифицируемыми в SQL. Критерии, по которым определяют, является ли представление модифицируемым или нет, в SQL, следующие [3]:

- Оно должно выводиться в одну и только в одну базовую таблицу.
- Оно должно содержать первичный ключ этой таблицы (это технически не предписывается стандартом ANSI, но было бы неплохо придерживаться этого).
- Оно не должно иметь никаких полей, которые бы являлись агрегатными функциями.
- Оно не должно содержать DISTINCT в своем определении.
- Оно не должно использовать GROUP BY или HAVING в своем определении.
- Оно не должно использовать подзапросы (это – ANSI ограничение которое не предписано для некоторых реализаций)
- Оно может быть использовано в другом представлении, но это представление должно также быть модифицируемыми.
- Оно не должно использовать константы, строки, или выражения значений среди выбранных полей вывода.
- Для INSERT, оно должно содержать любые поля основной таблицы которые имеют ограничение NOT NULL, если другое ограничение по умолчанию, не определено.

Правила, определяющие, является ли данное представление модифицируемыми в SQL, базируются на том принципе, что модификации в представлениях допустимы только когда SQL может недвусмысленно определить, какие значения базовой таблицы можно изменять.

Модифицируемые представления, в основном, используются точно так же, как и базовые таблицы. Фактически, пользователи не могут даже осознать, является ли объект, который они запрашивают, базовой таблицей или представлением. Это механизм защиты, предназначенный для сокрытия частей таблицы, которые являются конфиденциальными или не относятся к потребностям данного пользователя.

Представления «только чтение», с другой стороны, позволяют получать и переформатировать данные более рационально. Они являют собой набор сложных запросов, которые пользователь может выполнить неоднократно, сохраняя полученную информацию.

При решении поставленной задачи актуальной является возможность вставки данных для представлений.

При рассмотрении коммерческих и свободно распространяемых аналогов для генерации тестовых данных (Toad for SQL Server, EMS DataGenerator, Sybase Power Designer, Redgate SQL Data Generator, dbForge Studio for SQL Server) не было выявлено у них возможности генерации данных для представлений.

1.4 Ограничения целостности данных

Объекты реального мира представляются в реляционной базе данных как кортежи некоторого отношения. При разработке базы данных в неё умышленно вносятся так называемые «ограничения целостности».

Целостность данных – это механизм поддержания соответствия базы данных предметной области. В реляционной модели данных определены два базовых требования обеспечения целостности:

- целостность сущностей;
- целостность ссылок.

Требование целостности сущностей заключается в следующем: каждый кортеж любого отношения должен отличаться от любого другого кортежа этого отношения (т.е. любое отношение должно обладать первичным ключом).

Вполне очевидно, что если данное требование не соблюдается (т.е. кортежи в рамках одного отношения не уникальны), то в базе данных может храниться противоречивая информация об одном и том же объекте. Поддержание целостности сущностей обеспечивается средствами системы управления базой данных (СУБД). Это осуществляется с помощью двух ограничений:

- при добавлении записей в таблицу проверяется уникальность их первичных ключей;
- не допускается изменение значений атрибутов, входящих в первичный ключ.

Сложные объекты реального мира представляются в реляционной базе данных в виде кортежей нескольких нормализованных отношений, связанных между собой. При этом:

1. Связи между данными отношениями описываются в терминах функциональных зависимостей.
2. Для отражения функциональных зависимостей между кортежами разных отношений используется дублирование первичного ключа одного отношения (родительского) в другое (дочернее). Атрибуты, представляющие собой копии ключей родительских отношений, называются внешними ключами.

Требование целостности по ссылкам состоит в следующем: для каждого значения внешнего ключа, появляющегося в дочернем отношении, в родительском отношении должен найтись кортеж с таким же значением первичного ключа.

Как правило, поддержание целостности ссылок также возлагается на систему управления базой данных. Например, она может не позволить пользователю добавить запись, содержащую внешний ключ с несуществующим (неопределенным) значением [4].

Одним из требований к разрабатываемой системе является поддержка генерации данных с учётом ограничений целостности целевой БД.

1.5 Автоматизированный подбор стартовых параметров

Для решения задачи генерации данных в разрабатываемой системе предусмотрен набор модулей, называемых «генераторами данных». Каждый из генераторов предназначен для предоставления конечному пользователю функционала по генерации определенного типа данных и настройке определенных параметров генерации.

В то же время, реальная БД может состоять из нескольких сотен таблиц, для каждой из которых пользователю будет необходимо выбрать типы генерируемых данных и настроить параметры генерации. Программа может автоматически выбрать тип генерируемых данных, исходя из описания атрибута в целевой БД, но вопрос о «смысловой» составляющей данных должен решать в конечном итоге сам пользователь. Кроме того, логика программы по авто-выбору «генератора» для конкретного атрибута сущности БД, предопределенная разработчиком, может отличаться от логики, которой будет руководствоваться конечный пользователь системы.

В связи с этим, была поставлена задача: внедрить в программную систему модуль, отвечающий за автоматизированный подбор стартовых параметров генерации данных, который будет иметь возможность ручной настройки в соответствии с пользовательскими предпочтениями.

2. ОПИСАНИЕ ПРОГРАММНОЙ РЕАЛИЗАЦИИ

2.1 Декомпозиция задачи

В рамках поставленной цели были выделены следующие задачи при разработке программной системы:

- Проектирование пользовательского интерфейса программы.
- Реализация модулей для генерации данных различных типов.
- Реализация модуля по автоматизированному подбору стартовых параметров (АПСП) системы.
- Внедрение в разрабатываемую систему служебной базы данных для хранения системных данных и пользовательских настроек.
- Добавление в разрабатываемую систему логики для поддержки ограничений целостности целевой БД.
- Реализация функционала генерации данных для представлений.
- Реализация модуля для настройки пользовательских генераторов данных.

2.2 Архитектура программной системы

Архитектура проектируемой системы показана на рис. 7.

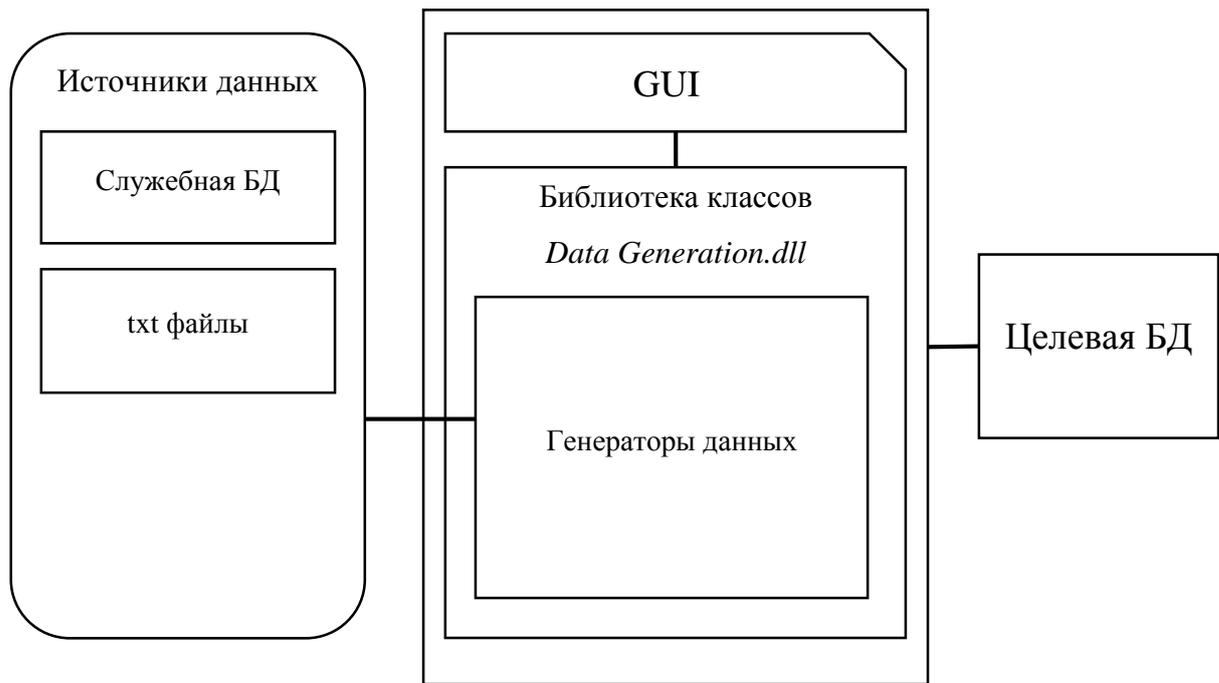


Рис. 7. Архитектура системы

Классы библиотеки *DataGeneration* проектируемой системы представлены в виде UML-диаграммы на рис. 8.

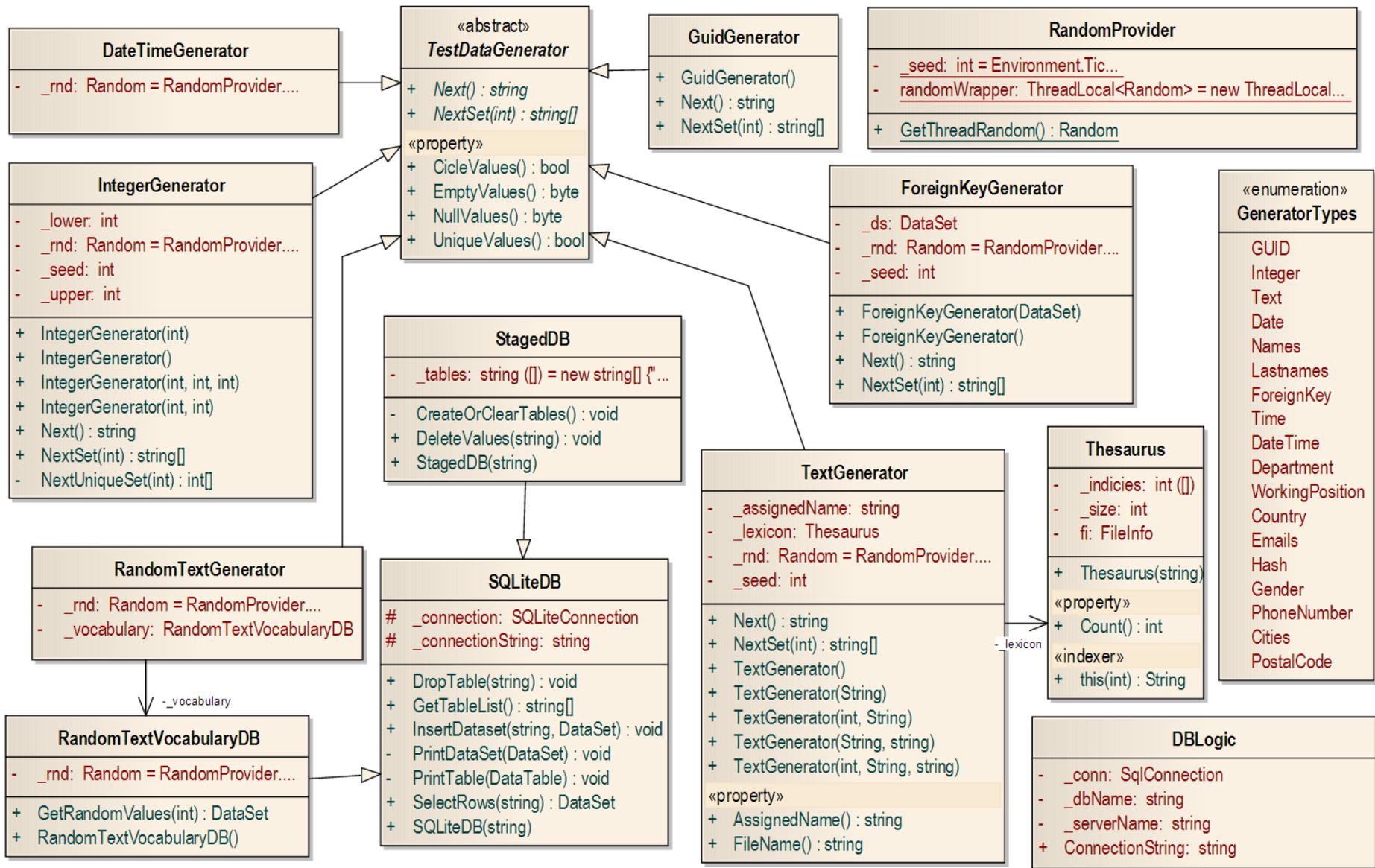


Рис. 8. Модель взаимодействия классов разрабатываемого решения в виде UML-диаграммы

Краткое описание классов:

- класс DBLogic – отвечает за взаимодействие с целевой БД,
- класс StagedDB – отвечает за взаимодействие со служебной БД,
- класс TestDataGenerator – предоставляет базовую структуру для модулей генерации данных,
- классы TextGenerator, RandomTextGenerator, DateTimeGenerator, GuidGenerator, IntegerGenerator, ForeignKeyGenerator – отвечают за генерацию различных типов данных,
- класс RandomProvider – предоставляет доступ к функциям генерации псевдослучайных чисел.

2.3 Пользовательский интерфейс

Оконный графический интерфейс разрабатываемой системы построен с использованием Windows Forms application programming interface (API) – интерфейс программирования приложений, отвечающий за графический интерфейс пользователя и являющийся частью Microsoft.NET Framework. Библиотека Windows Forms была разработана как часть .NET Framework для упрощения разработки компонентов графического интерфейса пользователя. Windows Forms построена на основе устаревающего Windows API и представляет собой, по сути, обертку низкоуровневых компонентов Windows [10].

Проект главного окна интерфейса пользователя представлен на рис. 9. Номерами 1-4 выделены основные области главного окна программы. Пользователю будет виден список таблиц и столбцов, которые можно отметить для генерации; настройки генерации, вид которых будет зависеть от выбранного генератора; предварительный просмотр сгенерированных данных. В главном меню программы пользователь сможет получить доступ к настройкам текущего проекта, возможности сохранения и загрузки последнего.

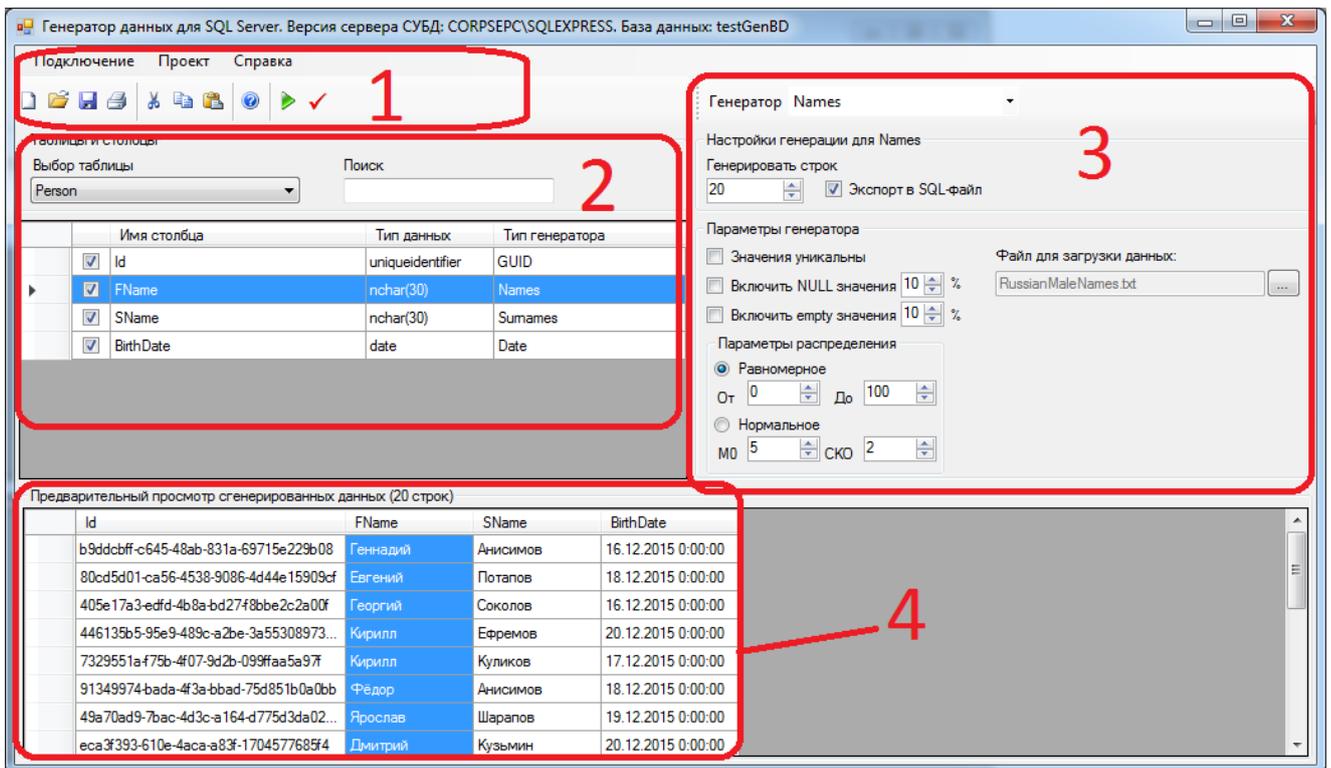


Рис. 9. Проект главного окна программы. Области: 1 – главное меню и панель инструментов, 2 – выбор таблиц и столбцов, 3 – настройки генерации, 4 – область предварительного просмотра

2.4 Примеры работы программы

На рис. 10 показаны результаты работы программы для таблицы Person, состоящей из четырех столбцов с типами uniqueidentifier, nchar, nchar, date.

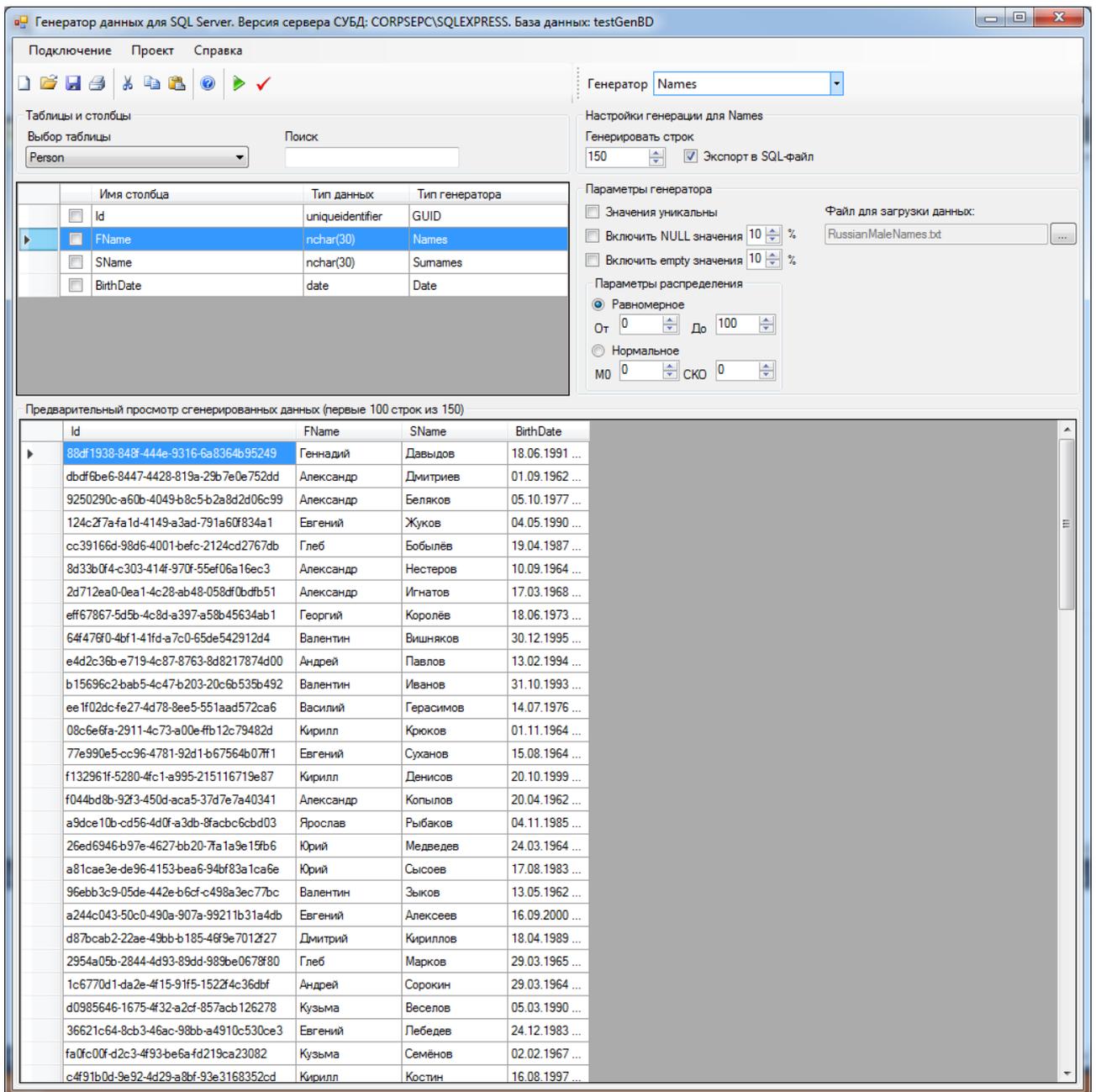


Рис. 10. Результаты работы программы для генераторов GUID, Names, Last-names, Date с параметрами по умолчанию

На рис. 11 показана ситуация, в которой пользователь включил режим генерации уникальных значений для столбца FName, тогда как множество значений выборки из файла заведомо меньше требуемого объема генерации.

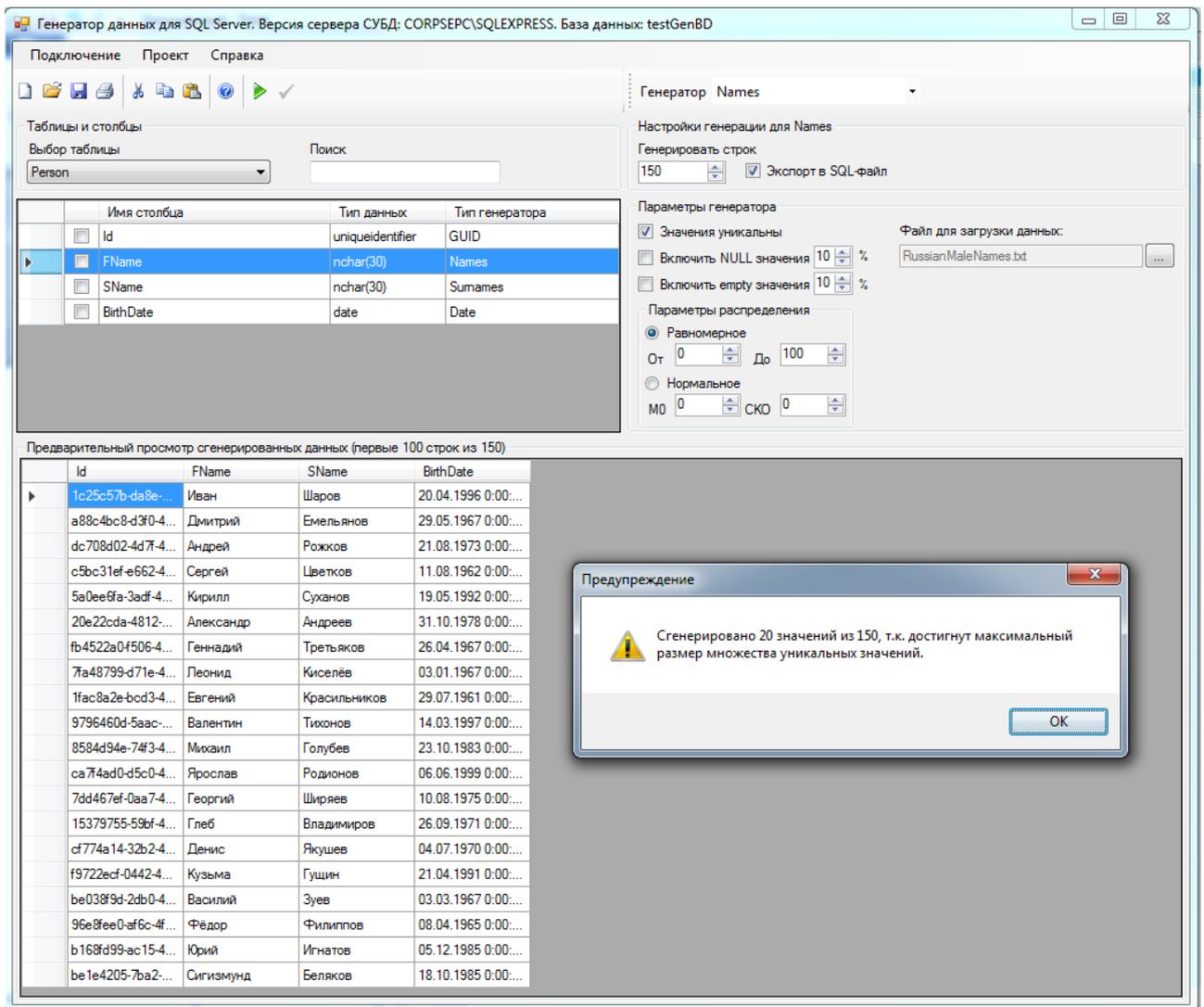


Рис. 11. Обработка исключительной ситуации

На рис. 12 показан случай, когда выставлен режим генерации с включением пустых значений в количестве 20% от общего объема генерации.

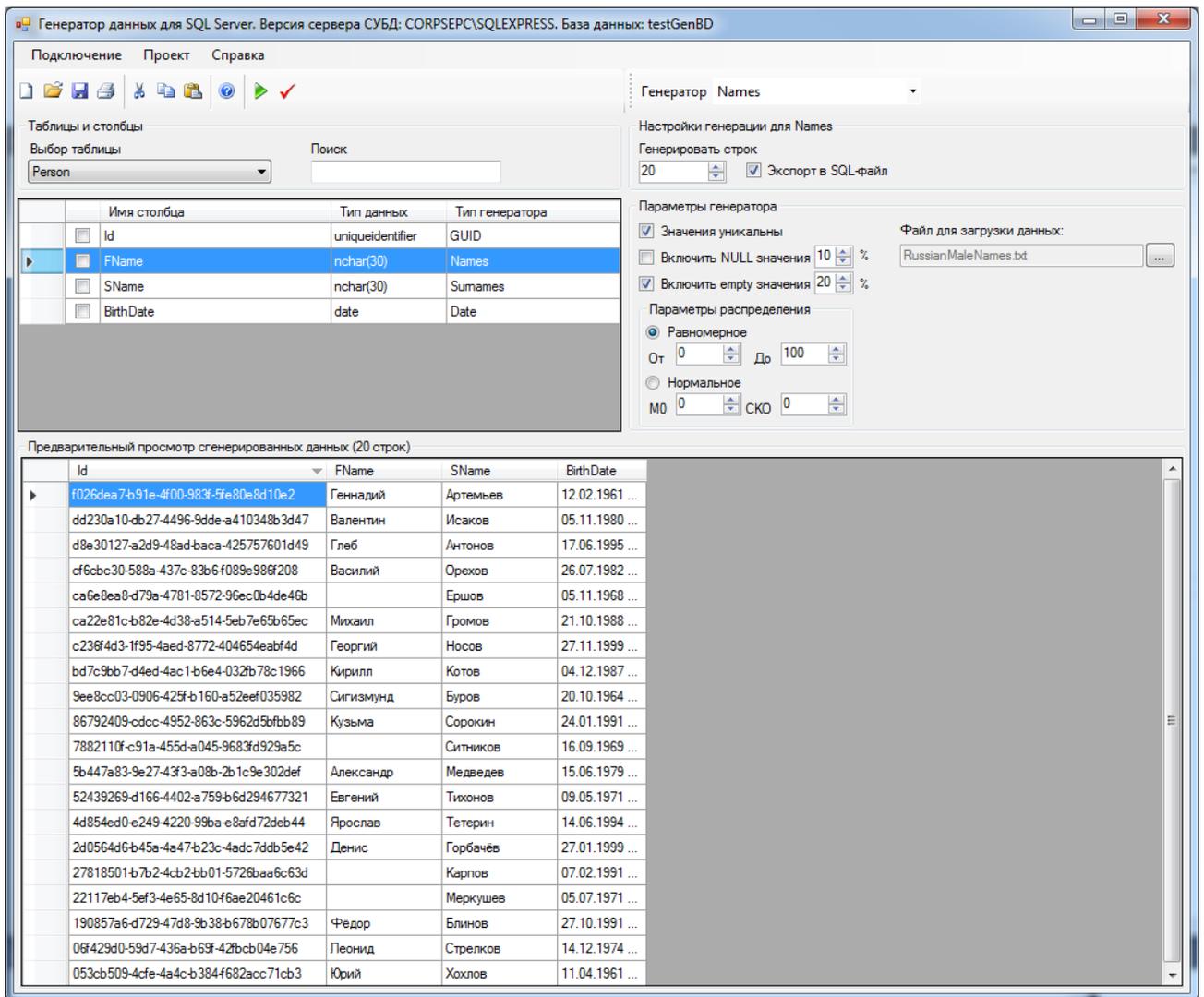


Рис. 12. Результат работы при выставлении пользователем параметра «Включить empty значения»

На рис. 13 показана работа программы для таблицы с столбцами типа Int, varchar, nvarchar. Для двух последних столбцов назначен генератор Text, который выдает последовательность случайных слов.

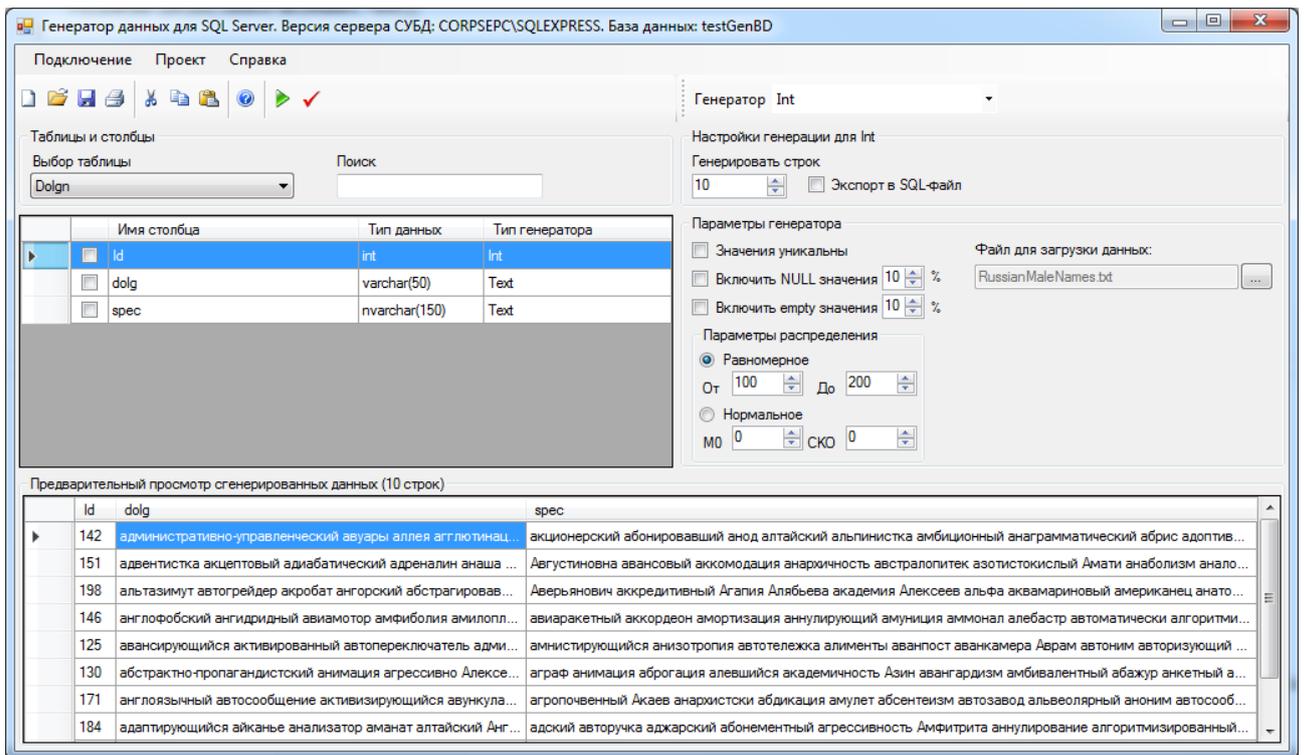


Рис. 13. Результаты работы программы для генераторов Int и Text

Пример файла, получаемого в результате работы модуля экспорта показан на рис. 14.

Id	FName	SName	BirthDate
7bce0aef-cec8-47dc-a77d-b3c40a828a15	Иван	Калинин	22.08.1976 0:00:00
6d13cdeb-15d8-4767-9f84-0726929ccc6d	Сигизмунд	Буров	26.02.1993 0:00:00
b1d4780f-d368-423c-ac4c-f77c850715bb	Денис	Яковлев	29.01.1961 0:00:00
b643ea8e-bf6a-4a6d-82a2-56d5ad2a1a60	Сергей	Агафонов	20.12.1960 0:00:00
5e362dcf-2cd0-4b87-b17b-86eeafe2f95c	Валентин	Рожков	23.03.1982 0:00:00
fd6cbf0c-e6c1-48f9-8511-8261d01983ae	Георгий	Устинов	12.06.1988 0:00:00
f6a03c0e-07c0-46b0-9b89-88b10be462aa	Кирилл	Виноградов	31.05.1978 0:00:00
4b1298c1-c616-4ea2-a385-83ef191a29c3	Георгий	Панов	15.09.1980 0:00:00
1effe184-9b50-4378-ae57-01804902d185	Сергей	Колобов	20.06.1993 0:00:00
8ae86ed8-a1ef-4f5c-b21b-d5db01795694	Валентин	Коновалов	14.02.1985 0:00:00
b0e42eb0-7576-462d-87a7-cd11f18a8706	Георгий	Пахомов	27.01.1996 0:00:00
2c65e59f-f509-4b67-83a3-4e381cf8e421	Иван	Моисеев	11.07.1981 0:00:00
2f47ad59-1625-4b63-94d1-254f77419ce0	Фёдор	Кошелев	04.01.1975 0:00:00
e2dd3ad6-1395-4a95-ae43-86daf4c4b2a7	Кирилл	Баранов	21.11.1970 0:00:00
6b5c3cef-a83b-45c7-8d5c-674927857a84	Фёдор	Носов	10.12.1994 0:00:00
fb247ad2-a4ff-4943-a84a-2fc07eecd546	Василий	Степанов	15.10.1997 0:00:00
eb8f2d00-a9c7-4229-ab25-c571700029fb	Денис	Копылов	26.11.1999 0:00:00
d99c02af-cd62-4748-b23e-5c380c4661ef	Валентин	Зайцев	14.01.1989 0:00:00
5fbc0141-a32f-48d5-8d67-86cfbca86625	Валентин	Третьяков	10.05.1979 0:00:00
a3eeb9a1-45a4-489c-835c-c4f778b6fb7d	Александр	Лукин	03.11.1984 0:00:00

GeneratedSQLQuery 26.10.2016...51))

```

INSERT INTO [testGenBD].[dbo].[Person]
([Id], [FName], [SName], [BirthDate])
VALUES
('7bce0aef-cec8-47dc-a77d-b3c40a828a15', 'Иван', 'Калинин', '22.08.1976 0:00:00')

INSERT INTO [testGenBD].[dbo].[Person]
([Id], [FName], [SName], [BirthDate])
VALUES
('6d13cdeb-15d8-4767-9f84-0726929ccc6d', 'Сигизмунд', 'Буров', '26.02.1993 0:00:00')

INSERT INTO [testGenBD].[dbo].[Person]
([Id], [FName], [SName], [BirthDate])
VALUES
('b1d4780f-d368-423c-ac4c-f77c850715bb', 'Денис', 'Яковлев', '29.01.1961 0:00:00')

INSERT INTO [testGenBD].[dbo].[Person]
([Id], [FName], [SName], [BirthDate])
VALUES
('b643ea8e-bf6a-4a6d-82a2-56d5ad2a1a60', 'Сергей', 'Агафонов', '20.12.1960 0:00:00')

INSERT INTO [testGenBD].[dbo].[Person]
([Id], [FName], [SName], [BirthDate])
VALUES
('5e362dcf-2cd0-4b87-b17b-86eeafe2f95c', 'Валентин', 'Рожков', '23.03.1982 0:00:00')

INSERT INTO [testGenBD].[dbo].[Person]
([Id], [FName], [SName], [BirthDate])
VALUES
('fd6cbf0c-e6c1-48f9-8511-8261d01983ae', 'Георгий', 'Устинов', '12.06.1988 0:00:00')

INSERT INTO [testGenBD].[dbo].[Person]
([Id], [FName], [SName], [BirthDate])
VALUES
('f6a03c0e-07c0-46b0-9b89-88b10be462aa', 'Кирилл', 'Виноградов', '31.05.1978 0:00:00')

INSERT INTO [testGenBD].[dbo].[Person]
([Id], [FName], [SName], [BirthDate])
VALUES
('4b1298c1-c616-4ea2-a385-83ef191a29c3', 'Георгий', 'Панов', '15.09.1980 0:00:00')

INSERT INTO [testGenBD].[dbo].[Person]
([Id], [FName], [SName], [BirthDate])
VALUES
('1effe184-9b50-4378-ae57-01804902d185', 'Сергей', 'Колобов', '20.06.1993 0:00:00')

```

Рис. 14. Сгенерированные данные (вверху) и результат их экспорта (внизу)

2.5 Основные алгоритмы работы

Общий алгоритм работы программы в нотации UML в виде диаграммы деятельности (UML Activity Diagram) показан на рис. 15.

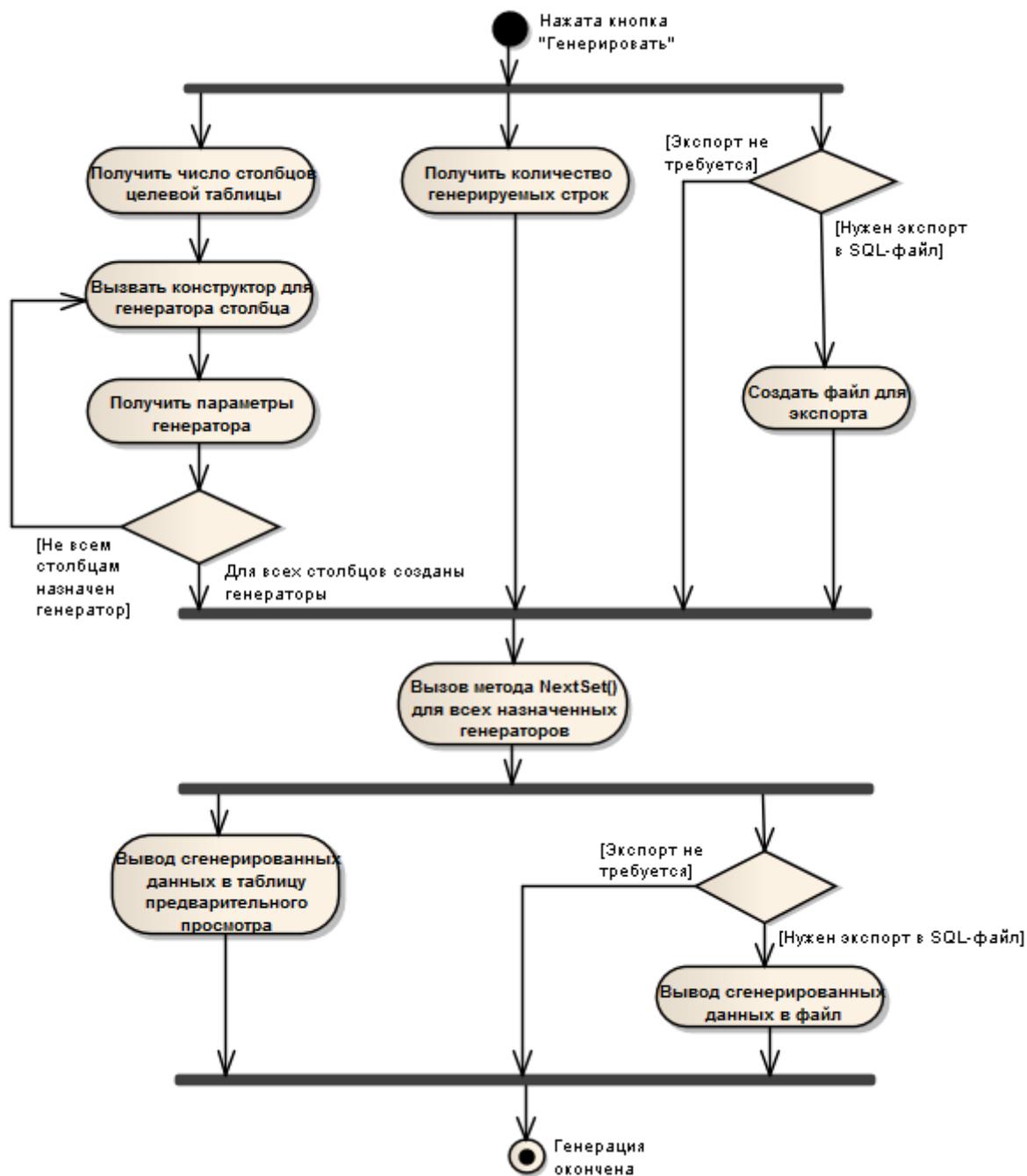


Рис. 15. Основной алгоритм работы программы

В качестве опционального параметра в пользовательском интерфейсе программы доступна возможность экспорта полученных в результате генерации значений в SQL-файл. При этом в подкаталоге Export корневого каталога

программы создается текстовый файл с расширением «sql». В данный файл при генерации каждого нового кортежа данных добавляется следующий текст:

```
«INSERT INTO [название базы данных].[имя пользователя].[название таблицы]
([список столбцов таблицы через запятую])
VALUES
('добавляемые значения через запятую')»
```

Алгоритм работы метода *NextSet()* генератора текстовых данных в виде UML-диаграммы деятельности показан на рис. 15.

Алгоритм основан на выборке случайных значений из заданного заранее множества слов и/или сочетаний слов.

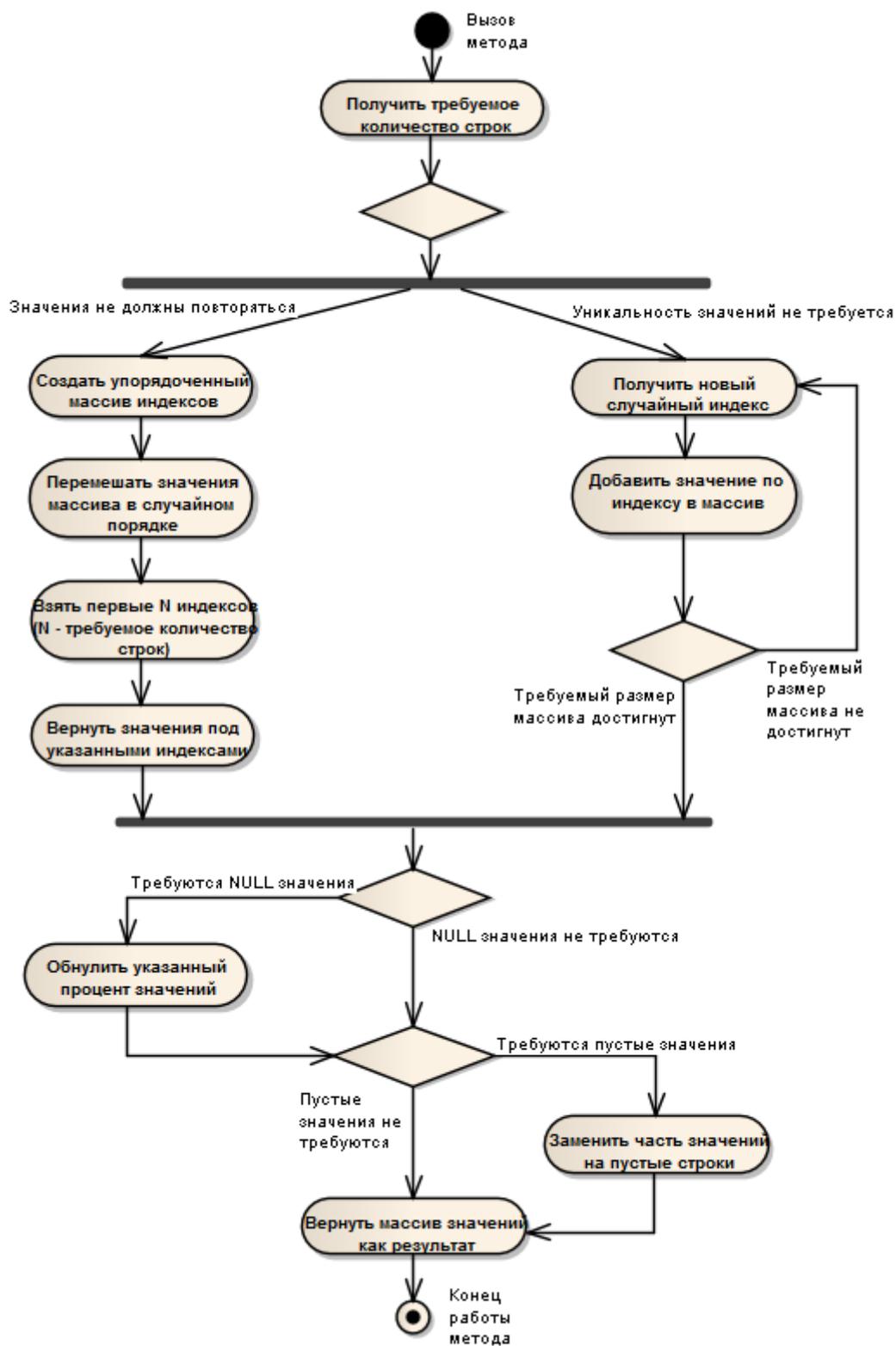


Рис. 16. Алгоритм работы текстового генератора данных

Алгоритм работы метода *NextSet()* генератора целочисленных данных типа «int» в виде UML-диаграммы деятельности показан на рис.17.

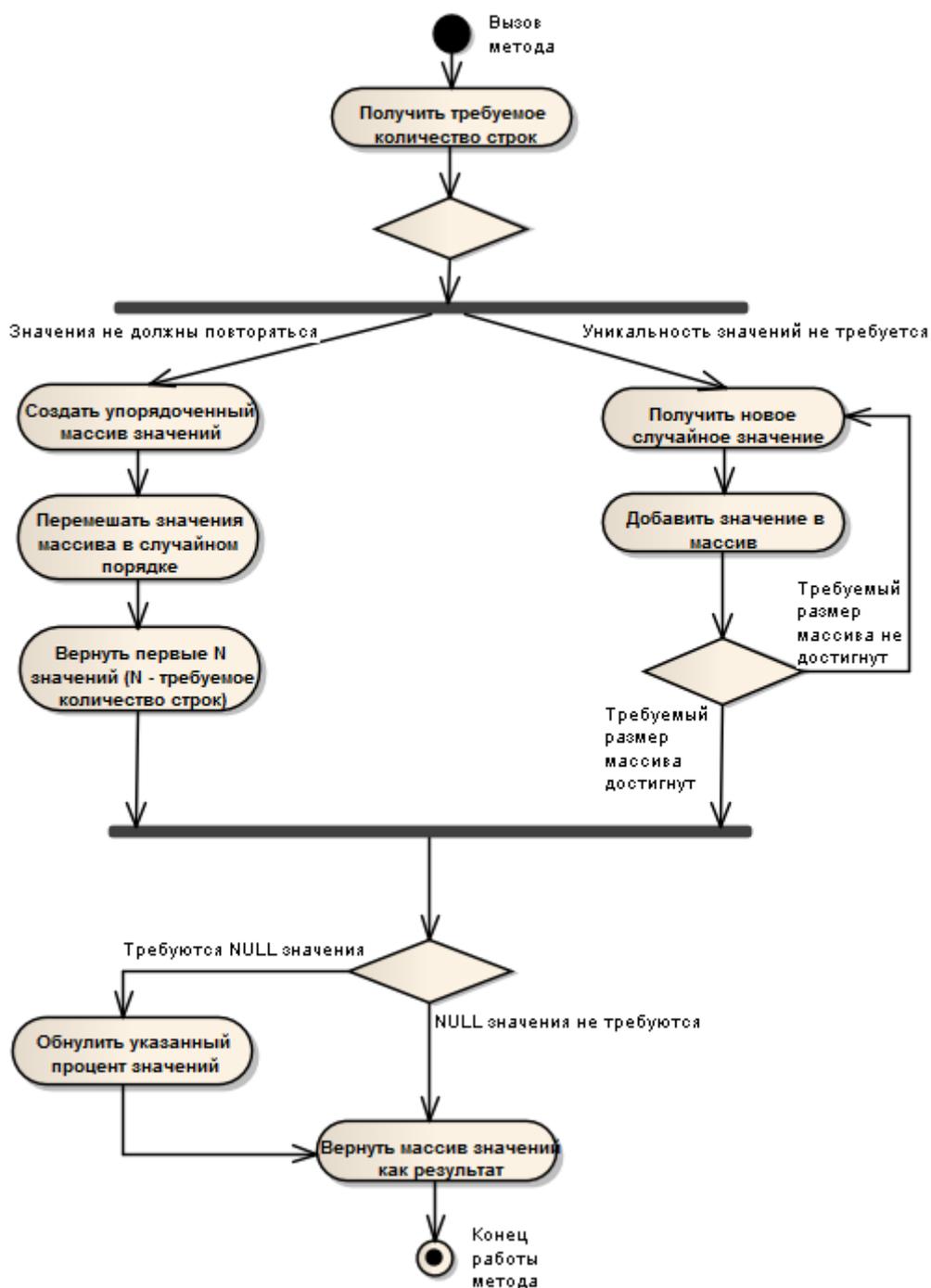


Рис. 17. Алгоритм работы целочисленного генератора данных

Алгоритм работы метода *NextSet()* генератора данных типа «Дата и время» в виде UML-диаграммы деятельности показан на рис. 18.

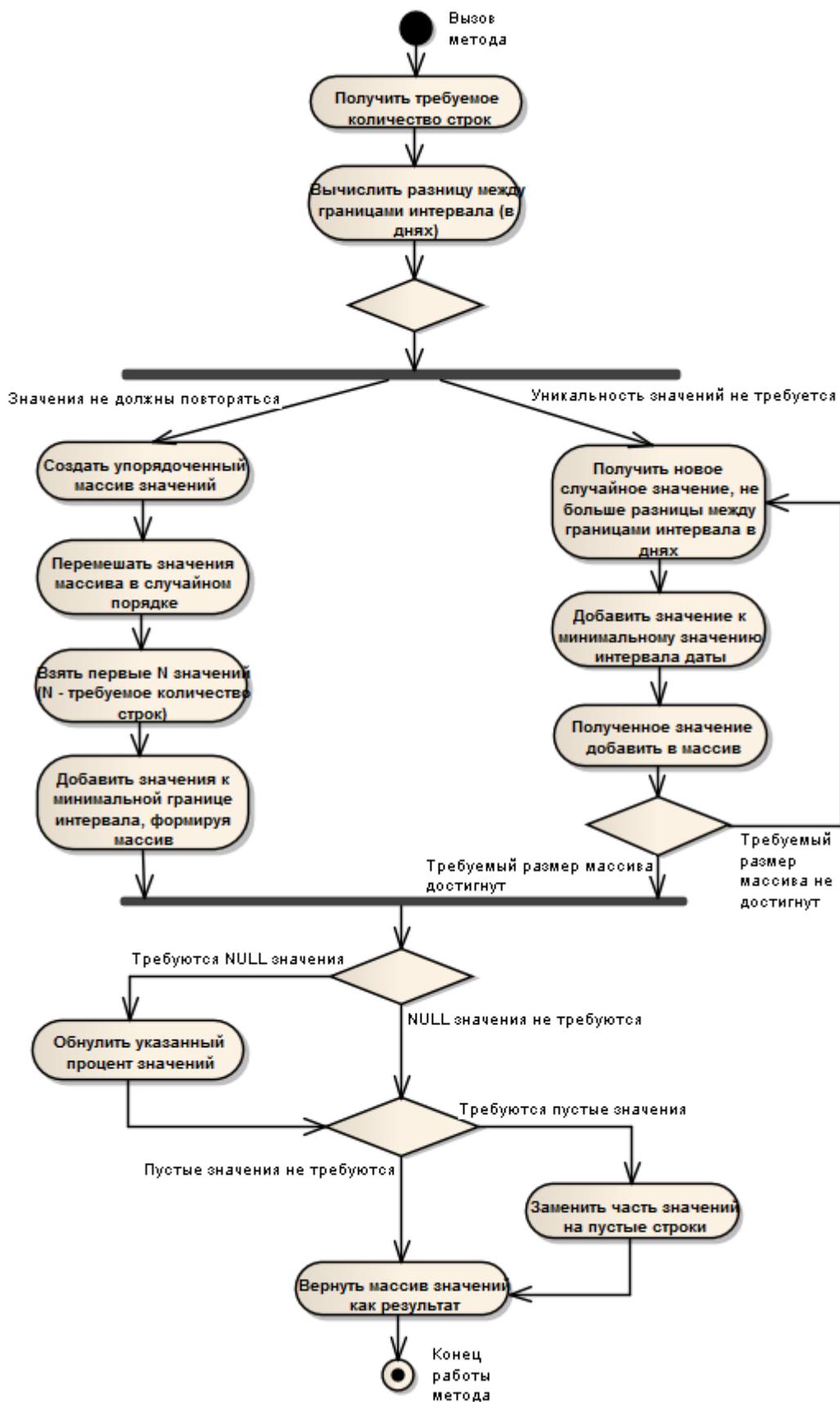


Рис. 18. Алгоритм работы генератора дат

Алгоритм работы генератора случайного текста в виде UML-диаграммы деятельности показан на рис. 19.

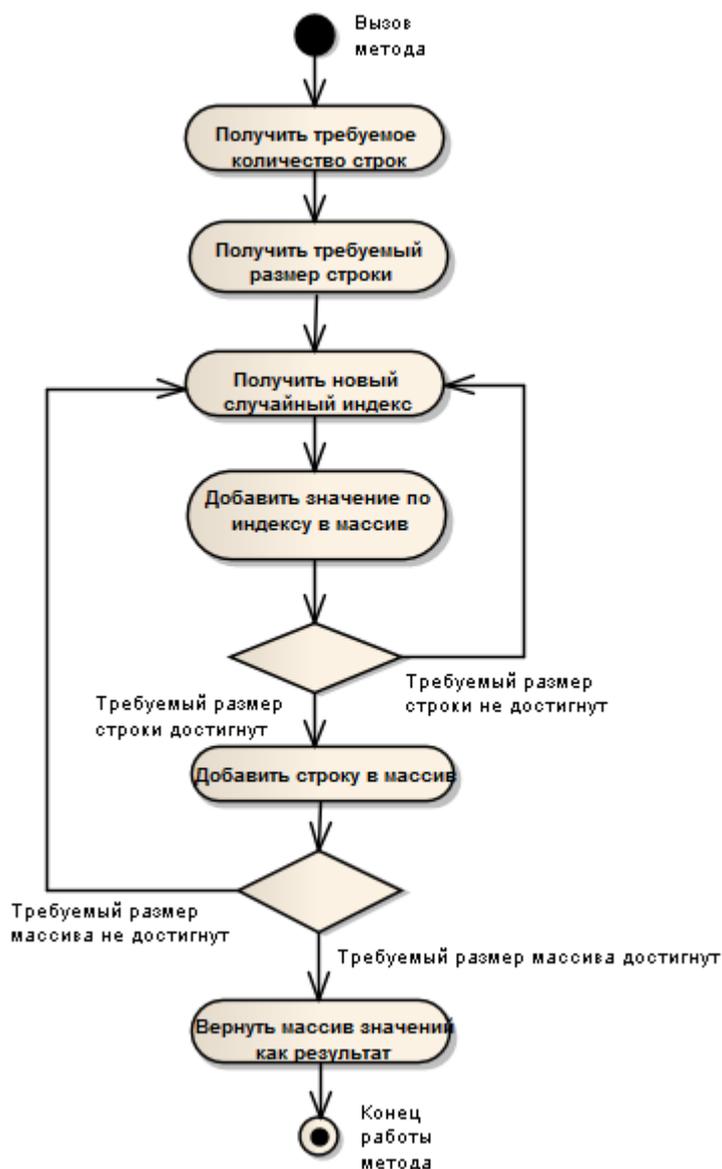


Рис. 19. Алгоритм работы генератора случайного текста

2.6 Служебная база данных

Цель использования в составе системы служебной БД – хранение пользовательских настроек и параметров программы.

Для управления служебной БД в составе программной системы была выбрана реляционная СУБД SQLite.

SQLite — компактная встраиваемая реляционная база данных. Исходный код библиотеки передан в общественное достояние. Слово «встраиваемый» (embedded) означает, что SQLite не использует парадигму клиент-сервер, то есть движок SQLite не является отдельно работающим процессом, с которым взаимодействует программа, а предоставляет библиотеку, с которой программа компонуется, и движок становится составной частью программы. Таким образом, в качестве протокола обмена используются вызовы функций (API) библиотеки SQLite [11].

Служебная БД имеет простую структуру, которая изображена на рис. 20.

Table Name	Columns	Indexes
FKList	FKName TEXT, TableName TEXT, ColumnName TEXT, RefTableName TEXT, RefColumnName TEXT	PRIMARY
TypeAutoAssignRules	Priority INT, GenType TEXT, Rule TEXT	PRIMARY
ColumnAttributes	ID INT, TABLE_NAME TEXT, COLUMN_NAME TEXT, DATA_TYPE TEXT, IS_NULLABLE TEXT, CHARACTER_MAXIMUM_LENGTH INT	PRIMARY
CustomGeneratorsList	CGName TEXT, Path TEXT	PRIMARY

Рис. 20. Таблицы служебной БД

Таблицы БД имеют следующее назначение:

- FKList – хранение списка внешних ключей.
- TypeAutoAssignRules – хранение правил авто-назначения генераторов.
- ColumnAttributes – хранение свойств столбцов таблиц.
- CustomGeneratorsList – хранение списка пользовательских генераторов данных и путей к файлам со значениями этих генераторов.
- RussianWordsDictionary – хранение объёмного списка слов русского языка.

2.7 Соблюдение ограничений целостности целевой БД

Целостность генерируемых данных в терминах сущностей реляционной модели данных в разрабатываемой системе сводится к требованию уникальности атрибута первичного ключа.

В разрабатываемой системе реализован механизм генерации уникальных значений через выставление свойств генераторов данных. Таким образом, задача сводится к тому, чтобы не допустить генерации повторяющихся значений путём программного присвоения определенного значения соответствующему свойству объекта генератора данных.

Схема алгоритма поведения системы при работе с данным ограничением для отдельно взятой таблицы представлена на рис. 21.



Рис. 21. Алгоритм работы с ограничениями целостности сущностей БД

Для ссылок требование целостности сводится к задаче ограничения возможного набора значений атрибута дочернего отношения. В родительском

отношении обязательно должен найтись кортеж с таким же значением ключевого атрибута.

В программной системе должна быть реализована логика, позволяющая системе достигать своей цели, не нарушая данного ограничения целостности.

Схема алгоритма поведения системы при работе с данным ограничением показана на рис. 22.

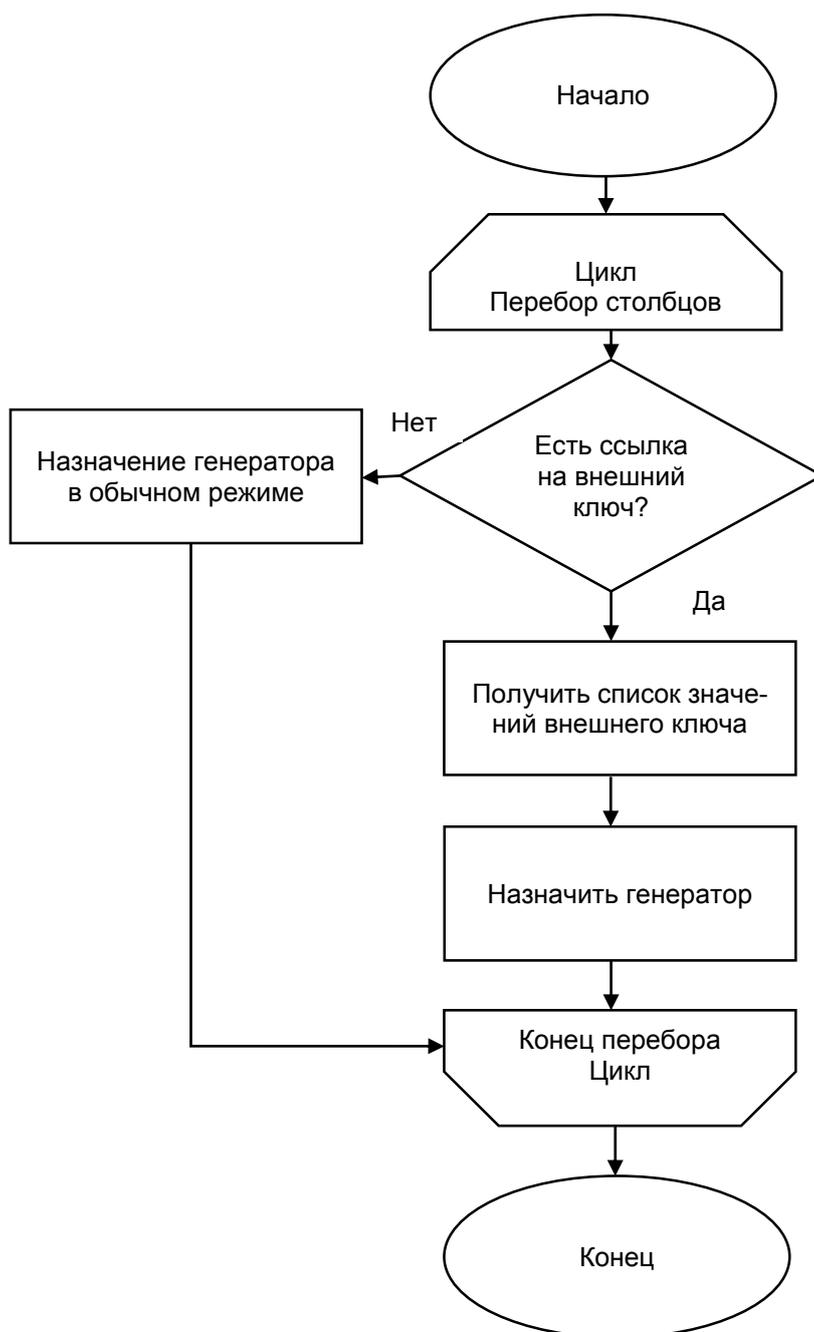


Рис. 22. Алгоритм работы программы при назначении генератора по внешнему ключу

2.8 Модуль автоматизированного подбора стартовых параметров

При проектировании пользовательского интерфейса модуля АПСР главной задачей было совместить возможность гибкой настройки правил автоназначения пользователем и простоту задания правил.

В результате был спроектирован интерфейс пользователя, представленный на рис. 23.

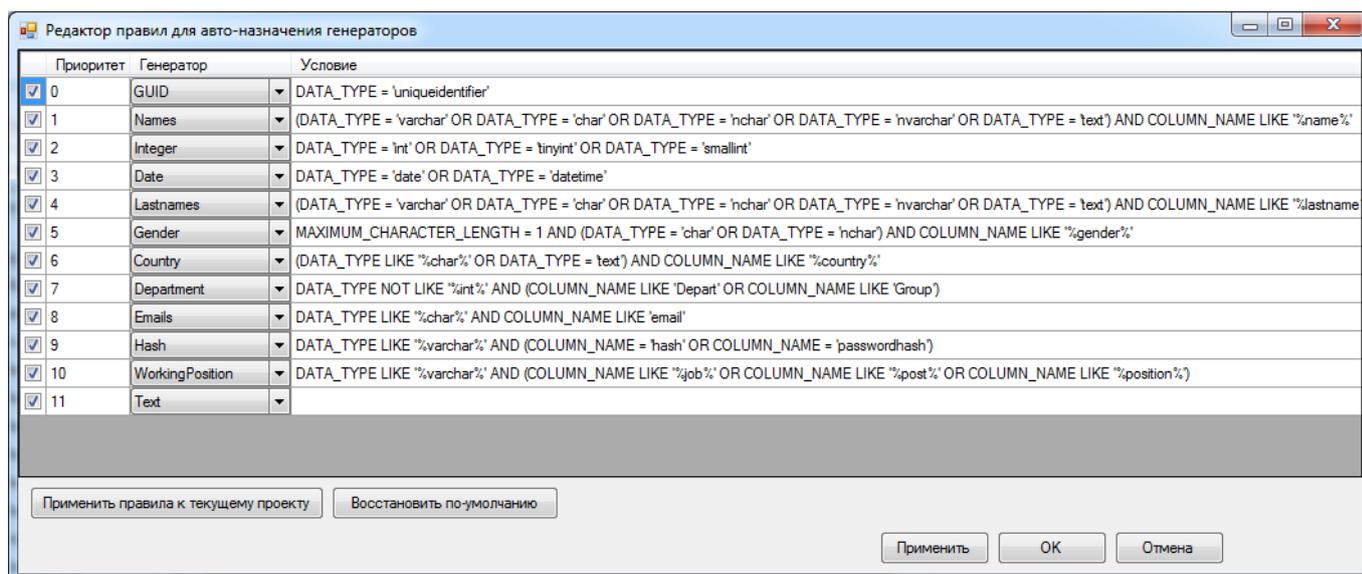


Рис. 23. Интерфейс редактора правил для авто-назначения генераторов

Главным элементом интерфейса пользователя является таблица, состоящая из 4 колонок. В первой колонке отображается статус текущего правила. Имеет 2 значения: включено – используется системой, и выключено – игнорируется системой. Вторая колонка показывает приоритет правила. Правила с более низким приоритетом располагаются выше в строках таблицы, чем правила с более высоким приоритетом. Очередность выполнения такова: сначала система применяет правила с более высоким приоритетом, затем – с более низким. Приоритет также служит для разрешения конфликтов, когда похожие условия соответствуют разным генераторам. В таких случаях будет применено правило, имеющее более низкий приоритет. Третий столбец – вид генератора, позволяет выбрать генератор данных из выпадающего списка. Четвертый столбец таблицы содержит непосредственно правила.

Для обеспечения гибкости настройки правил использован специализированный язык задания правил. В качестве такого языка был выбран язык, близкий к SQL.

Правила на этом языке имеют 4 основных параметра, каждый из которых обладает собственным ключевым словом. Список параметров и их ключевых слов приведен в таблице 2.

Таблица 2 - Параметры и ключевые слова для задания правил авто назначения генераторов

Параметр	Ключевое слово
Тип данных атрибута	DATA_TYPE
Имя столбца атрибута	COLUMN_NAME
Ограничение длины текстового поля	CHARACTER_MAXIMUM_LENGTH
Может ли принимать NULL значения	IS_NULLABLE

Для сравнения параметров с задаваемыми пользователем значениями могут использоваться любые операторы, аналогичные языку SQL. Кроме операторов сравнения также могут использоваться функции алгебры логики. Примеры используемых операторов приведены в таблице 3.

Таблица 3 - Операторы и связанные параметры

Оператор	Значение	Параметры, с которыми может изменяться
AND	Логическое «И»	-
OR	Логическое «Или»	-
=	Равно	Любые
!=	Не равно	Любые
>	Больше	CHARACTER_MAXIMUM_LENGTH
<	Меньше	CHARACTER_MAXIMUM_LENGTH
>=	Больше или равно	CHARACTER_MAXIMUM_LENGTH
<=	Меньше или равно	CHARACTER_MAXIMUM_LENGTH

LIKE	Устанавливает соответствие символьной строки с шаблоном	Любые, кроме CHARACTER_MAXIMUM_LENGTH
------	---	---------------------------------------

К примеру, правило для назначения генератора имён в базовом наборе правил задаётся следующим образом:

«(DATA_TYPE = 'varchar' OR DATA_TYPE = 'char' OR DATA_TYPE = 'nchar' OR DATA_TYPE = 'nvarchar' OR DATA_TYPE = 'text') AND COLUMN_NAME LIKE '%name%'»

Данная запись означает, что для атрибутов с типами «varchar», «char», «nchar», «nvarchar» или «text», имя которых содержит подстроку «name» будет автоматически выбираться генератор имён.

Базовый набор правил программы показан в таблице 8.

Таблица 4 - Базовый набор правил авто назначения генераторов

Генератор	Правило
GUID	DATA_TYPE = 'uniqueidentifier'
Names	(DATA_TYPE = 'varchar' OR DATA_TYPE = 'char' OR DATA_TYPE = 'nchar' OR DATA_TYPE = 'nvarchar' OR DATA_TYPE = 'text') AND COLUMN_NAME LIKE '%name%'
Integer	DATA_TYPE = 'int' OR DATA_TYPE = 'tinyint' OR DATA_TYPE = 'smallint'
Date	DATA_TYPE = 'date' OR DATA_TYPE = 'datetime'
Lastnames	DATA_TYPE LIKE 'char' AND COLUMN_NAME LIKE '%lastname%'
Gender	MAXIMUM_CHARACTER_LENGTH = 1 AND (DATA_TYPE = 'char' OR DATA_TYPE = 'nchar') AND COLUMN_NAME LIKE '%gender%'
Country	(DATA_TYPE LIKE '%char%' OR DATA_TYPE = 'text') AND COLUMN_NAME LIKE '%country%'
Department	DATA_TYPE NOT LIKE '%int%' AND (COLUMN_NAME LIKE 'Depart' OR COLUMN_NAME LIKE 'Group')
Emails	DATA_TYPE LIKE '%char%' AND COLUMN_NAME LIKE 'email'

Hash	DATA_TYPE LIKE '%varchar%' AND (COLUMN_NAME = 'hash' OR COLUMN_NAME = 'passwordhash')
WorkingPosition	DATA_TYPE LIKE '%varchar%' AND (COLUMN_NAME LIKE '%job%' OR COLUMN_NAME LIKE '%post%' OR COLUMN_NAME LIKE '%position%')
Text	

Пустая последняя строка в графе «Правило» означает что для всех атрибутов, которые не отвечают ни одному правилу, будет назначен генератор Text, отвечающий за генерацию случайного текста.

Пример работы модуля АПСИ для базового набора правил, показан на рис. 24.

	Имя столбца	Тип данных	Тип генератора
<input checked="" type="checkbox"/>	DepartmentID	smallint	Integer
<input checked="" type="checkbox"/>	Name	nvarchar(50)	Names
<input checked="" type="checkbox"/>	GroupName	nvarchar(50)	Names
<input checked="" type="checkbox"/>	ModifiedDate	datetime	Date

Рис. 24. Столбцы таблицы с типами данных и автоматически назначенными генераторами.

2.9 Проектирование интерфейса пользователя для модулей генерации данных

В целях экономии рабочего пространства главного окна пользовательского интерфейса программы был спроектирован интерфейс, позволяющий менять область изменения параметров генерации в зависимости от выбранного в данный момент генератора данных. На рис. 25 данная область выделена рамкой.

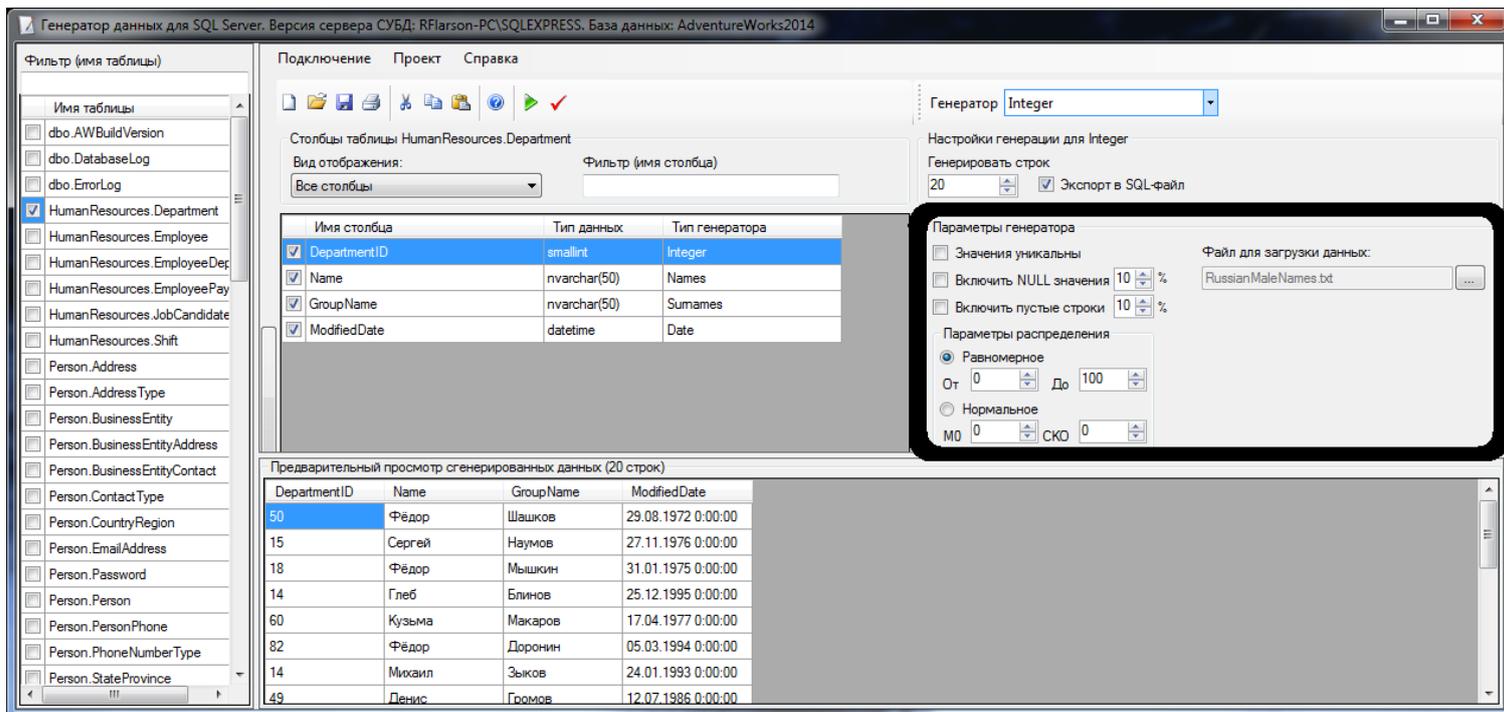


Рис. 25. Главное окно интерфейса пользователя. Область настройки параметров генерации выделена рамкой

Были спроектированы пользовательские интерфейсы для различных видов генераторов данных. Их внешний вид показан на рисунках 26 – 31.

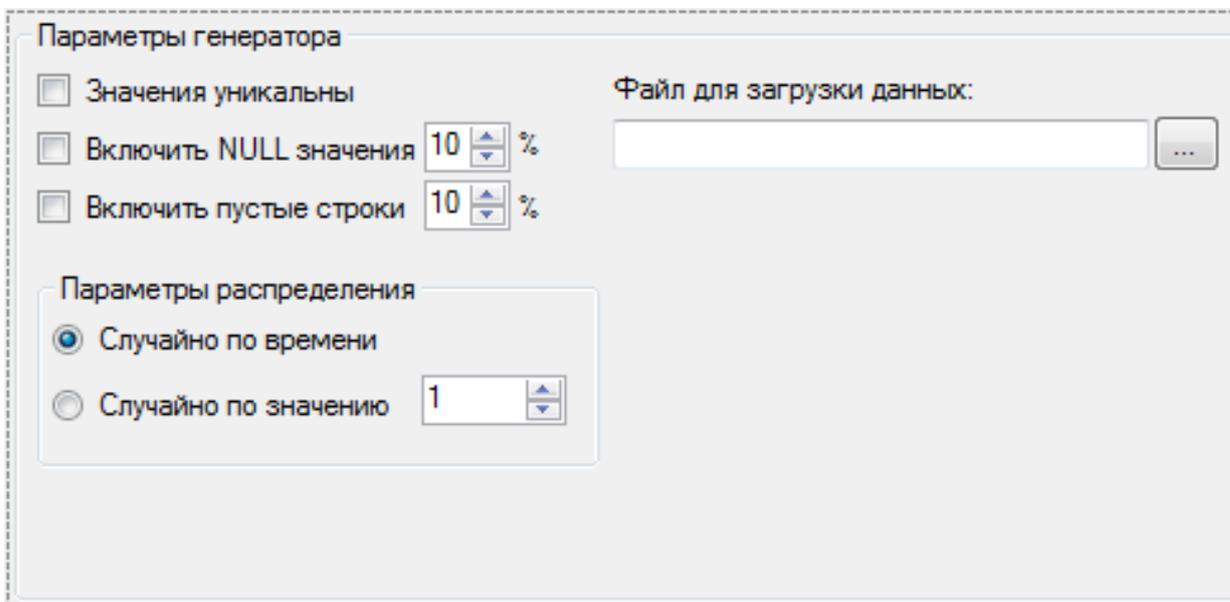


Рис. 26. Пользовательский интерфейс генератора текстовых данных

Параметры генератора

Значения уникальны

Включить NULL значения 10 %

Параметры распределения

Случайно по времени

Случайно по значению 1

Инкремент 1

Значения

Мин. 0

Макс. 4096

Рис. 27. Пользовательский интерфейс генератора целочисленных данных

Параметры генератора

Значения уникальны

Включить NULL значения 10 %

Параметры распределения

Случайно по времени

Случайно по значению 1

Инкремент

День 1

Время 0:00:00

Значения

	Дата	Время
Мин.	01.01.1970	0:00:00
Макс.	01.04.2017	0:00:00

Рис 28. Пользовательский интерфейс генератора даты и времени

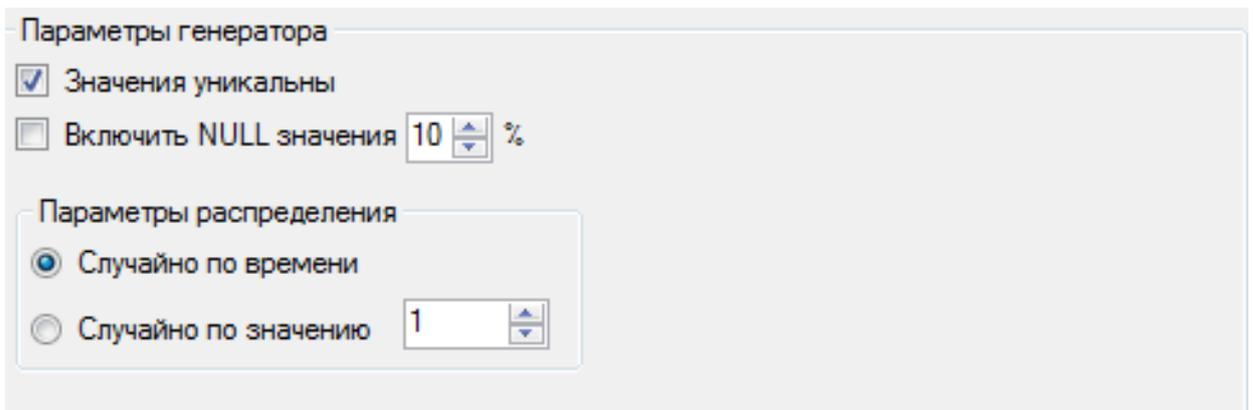


Рис. 29. Пользовательский интерфейс генератора Globally Unique Identifier (GUID)

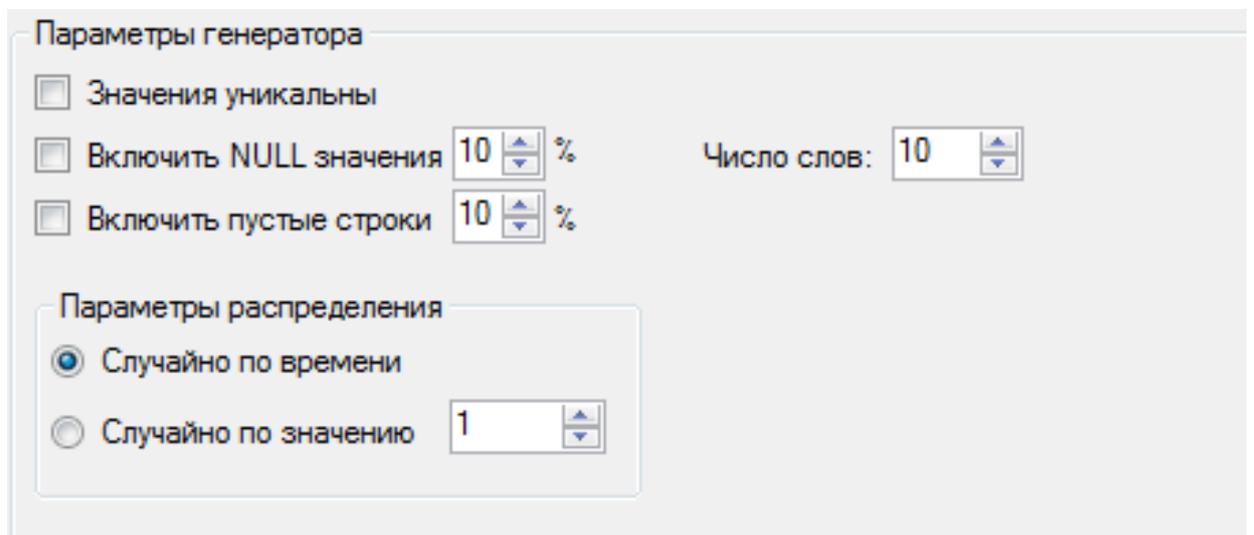


Рис. 30. Пользовательский интерфейс генератора случайного текста

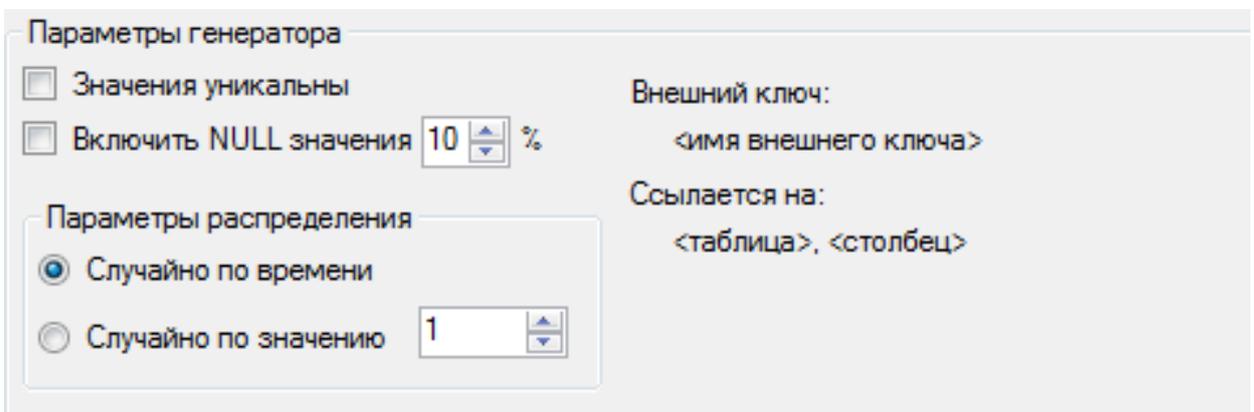


Рис. 31. Пользовательский интерфейс генератора по внешнему ключу

2.10 Генерация данных для представлений

Как было сказано в первой главе, для вставки данных в базовые таблицы через представления, последние должны отвечать критериям модифицируемости. В случае, если представление не является модифицируемым, попытка вставить в него данные вызовет исключение управляющей СУБД.

С помощью триггера `INSTEAD OF INSERT` можно задать проверку значений или ошибок одного или нескольких столбцов с последующим выполнением дополнительных действий перед вставкой записи.

Триггеры `INSTEAD OF INSERT` могут быть заданы в представлении или таблице с целью замены стандартного действия инструкции `INSERT`. Обычно триггер `INSTEAD OF INSERT` определяется в представлении для вставки данных в одну или несколько базовых таблиц [4].

Столбцы в списке выборки представления могут быть либо пустыми, либо нет. Если столбцу представления не разрешено иметь значения `NULL`, то инструкция `INSERT` должна определить значения для этого столбца. Столбцы представления могут иметь значение `NULL`, если в определяющем этот столбец выражении содержатся следующие элементы:

- ссылки на любой столбец базовой таблицы, допускающий значения `NULL`;
- арифметические операторы;
- ссылки на функции;
- `CASE` или `COALESCE` с подвыражением, которое может иметь значение `NULL`;
- `NULL IF`.

Таким образом, задачу генерации данных для представления можно свести к проверке существования триггера `INSTEAD OF INSERT` для данного представления в БД.

Схема алгоритма работы программы при генерации данных для представлений показана на рис. 32.

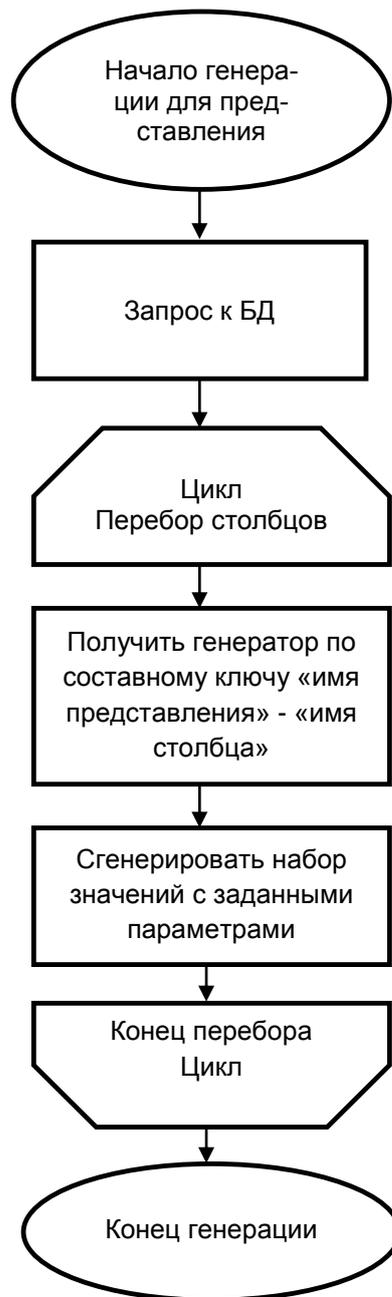


Рис. 32. Алгоритм генерации данных для представлений

В интерфейс программы были добавлены элементы для отображения представлений в списке таблиц. На рисунках 33-35 показаны различные режимы отображения списка таблиц с возможностью выбора.

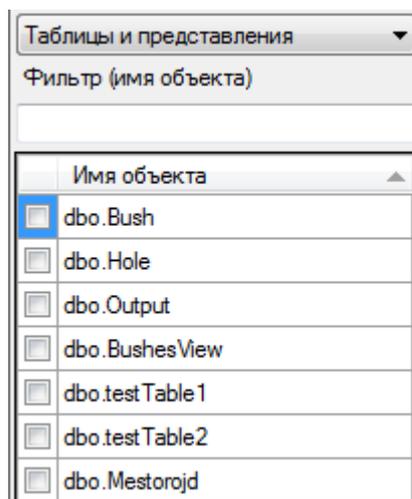


Рис. 33 Список объектов БД, режим отображения «Таблицы и представления»

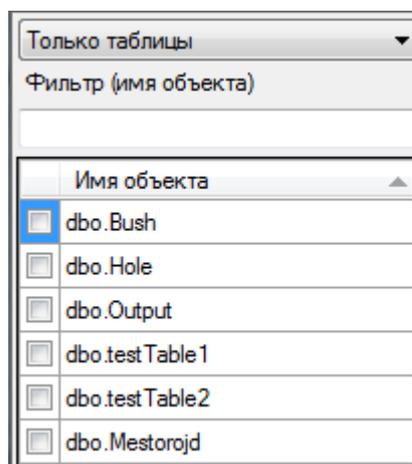


Рис. 34 Список объектов БД, режим отображения «Только таблицы»

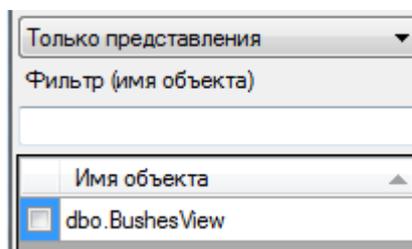


Рис. 35 Список объектов БД, режим отображения «Только представления»

Работа с представлениями через интерфейс пользователя ничем не отличается от работы с базовыми таблицами. Для них доступны те же функции настройки генерации, выбор типа генератора и его параметров. На рис. 36 показан интерфейс программы при работе с представлением «dbo.BushesView».

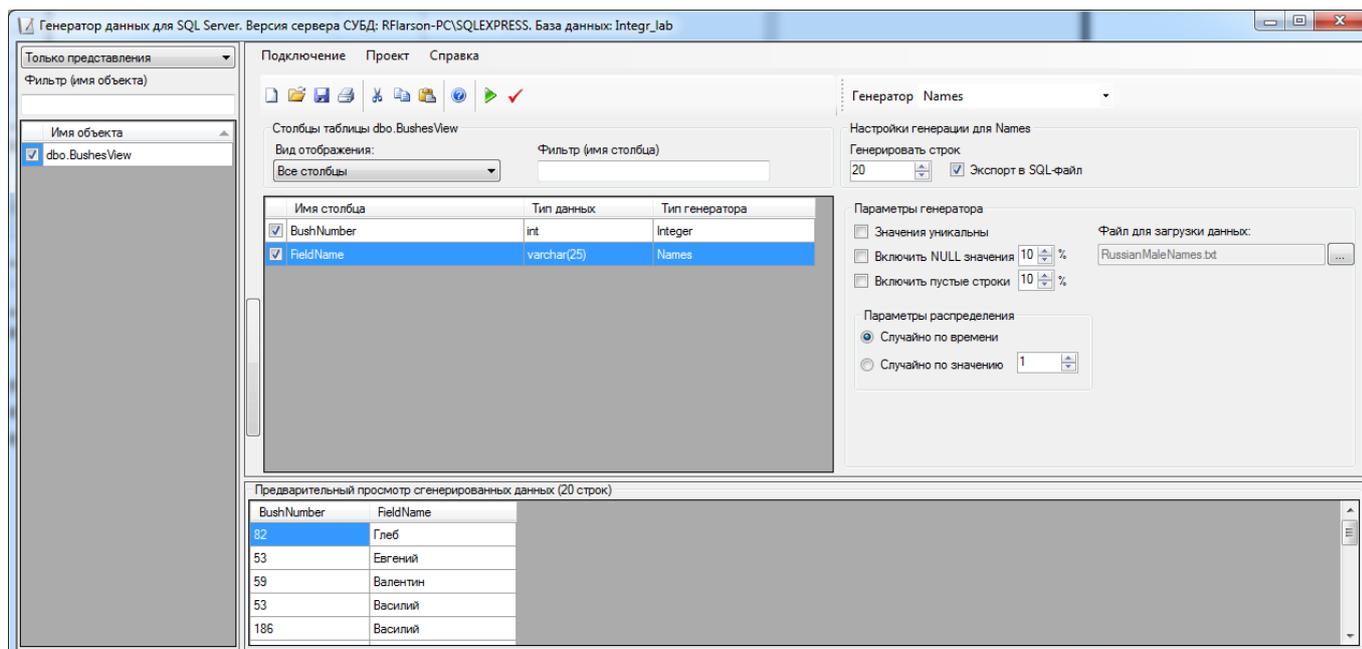


Рис. 36 Интерфейс программы при работе с представлениями

2.11 Модуль задания пользовательских типов генераторов

В составе системы по умолчанию предусмотрены следующие виды текстовых данных:

- Мужские имена,
- Мужские фамилии,
- Рабочие должности,
- Страны,
- Адреса электронной почты,
- Значения хэш-функций,
- Номера телефонов,
- Города,
- Почтовые индексы

Очевидно, что данный набор видов данных не в состоянии охватить большую часть различных тематических данных. Цель разработки данного модуля – предоставить пользователю возможность расширить функционал программы, затрачивая минимальные усилия.

Данный модуль предназначен для добавления и редактирования пользователем дополнительных генераторов текстовых данных.

Пользовательский интерфейс модуля показан на рис. 37.

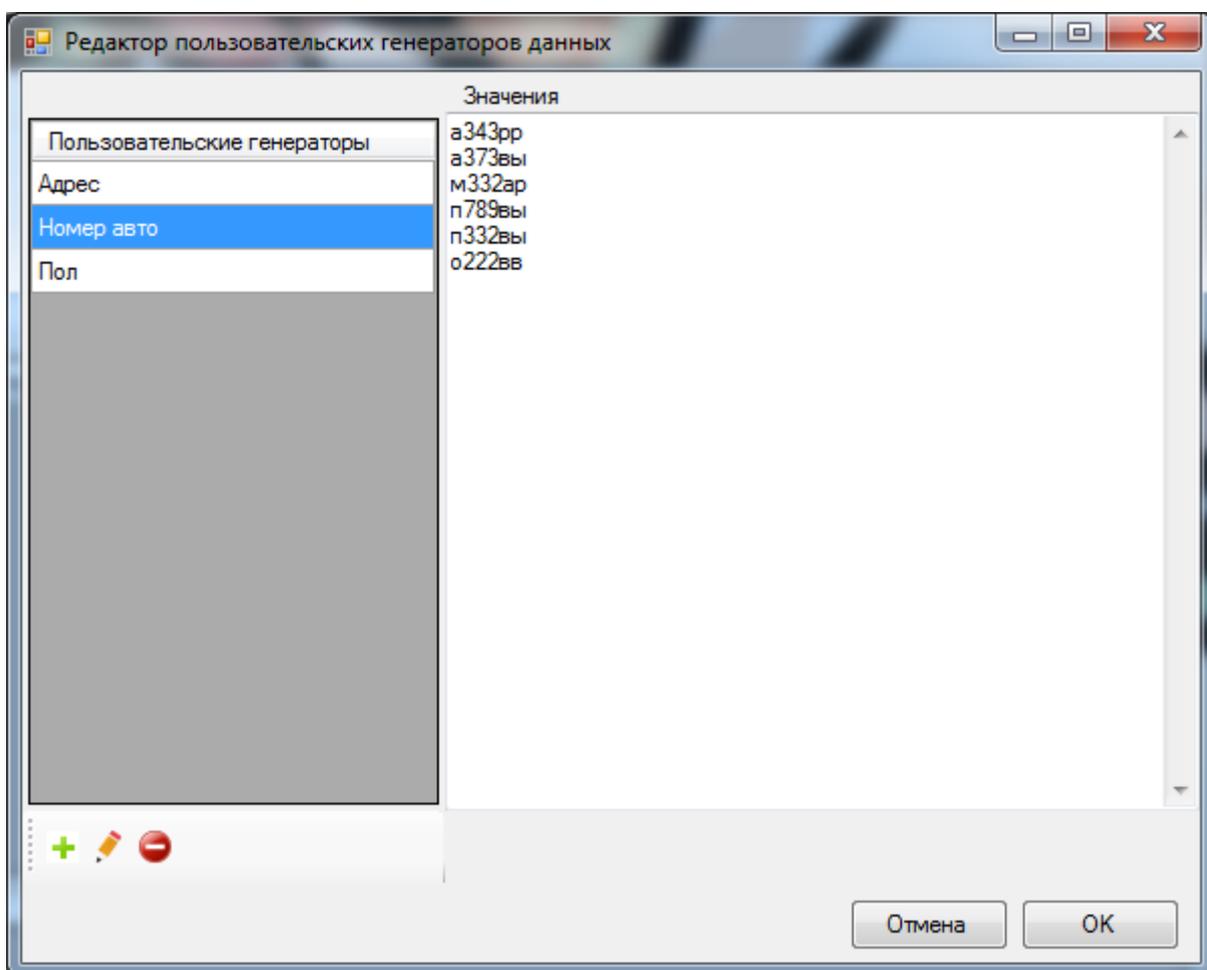


Рис. 37. Окно редактора пользовательских генераторов данных

В левой части окна находится список генераторов, заданных пользователем, в правой части окна отображается список значений для выделенного в данный момент генератора. Кнопки для добавления, редактирования и удаления новых генераторов находятся под списком генераторов.

2.12 Пример работы модуля пользовательских генераторов данных

Через интерфейс модуля по заданию пользовательских генераторов данных в программу были добавлены 3 новых генератора текстовых данных:

пол, номер авто, и адрес. В результате эти генераторы появились в списке доступных генераторов в главном окне пользовательского интерфейса (рис. 38).

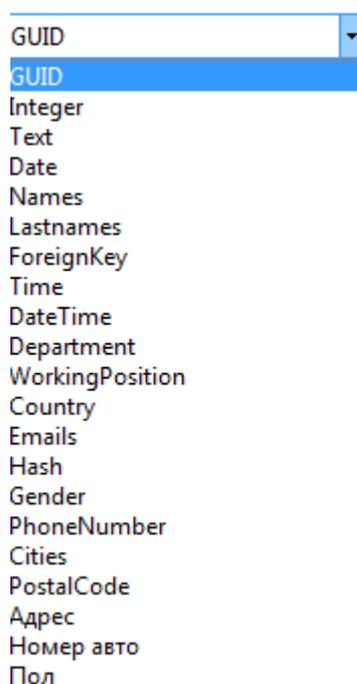


Рис. 38. Выпадающий список доступных генераторов данных, с добавлением заданных пользователем генераторов

На рис. 39 показан пример работы добавленных генераторов данных.

	Имя столбца	Тип данных	Тип генератора
<input type="checkbox"/>	ID	uniqueidentifier	Номер авто
<input type="checkbox"/>	Name	varchar(25)	Адрес
<input checked="" type="checkbox"/>	BushID	uniqueidentifier	Пол

Предварительный просмотр сгенерированных данных (20 строк)

ID	Name	BushID
м332ар	Рудокопровая, 7	Ж
а373вы	Рудокопровая, 7	М
о222вв	Ленина, 5	М
м332ар	Пирогова, 10	М
м332ар	Богдана Хмельницкого, пер., 12	М
п332вы	Богдана Хмельницкого, пер., 12	Ж
а343рр	Трудармейская, 22	М
о222вв	Пирогова, 10	Ж
м332ар	Строителей, 19	Ж
о222вв	Пирогова, 10	Ж
п789вы	Рудокопровая, 7	М
а373вы	Строителей, 19	Ж
о222вв	Ленина, 5	М

Рис. 39. Результаты работы добавленных пользователем генераторов данных

3. ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И РЕСУРСОСБЕРЕЖЕНИЕ

3.1. Оценка коммерческого и инновационного потенциала НТИ

3.1.1 Потенциальные потребители

Целевой рынок – рынок программного обеспечения (ПО).

На данном рынке можно по критерию целевого назначения выделить следующие области рынка [12]:

- 1) Системное ПО
 - 1.1) Операционные системы
 - 1.2) Базы данных, управление электронными данными
- 2) Информационная безопасность
 - 2.1) Антивирусное ПО
- 3) Сетевое, интернет и интранет ПО
- 4) Бизнес-ориентированное ПО
 - 4.1) Офисное ПО
 - 4.2) Распознавание текста
 - 4.3) Инструментальное
 - 4.4) Переводчики с иностранных языков
- 5) Коммуникационное и мультимедийное ПО

Целевым для проекта в данном случае рынком является инструментальное ПО, предназначенное для разработчиков программного обеспечения, баз данных (БД) и web-сайтов. Под разработчиками понимаются как физические, так и юридические лица (компании).

3.1.2 Анализ конкурентных технических решений с позиции ресурсоэффективности и ресурсосбережения

В качестве конкурирующих решений были выбраны следующие: K_1 – Toad for SQL Server, K_2 – Redgate SQL Data Generator, K_3 – Devart dbForge Studio for SQL Server.

Таблица 3.1 - Оценочная карта для сравнения конкурентных решений

Критерии оценки	Вес критерия	Баллы				Конкурентоспособность			
		B_{ϕ}	B_{K1}	B_{K2}	B_{K3}	K_{ϕ}	K_{K1}	K_{K2}	K_{K3}
1	2	3	4	5	6	7	8	9	10
Технические критерии оценки ресурсоэффективности									
1. Повышение производительности труда пользователя	0,1525	3	3	4	5	0,4575	0,4575	0,61	0,7625
2. Удобство в эксплуатации (соответствует требованиям потребителей)	0,098	3	3	4	5	0,294	0,294	0,392	0,49
3. Надежность	0,123	1	4	4	4	0,123	0,492	0,492	0,492
4. Потребность в ресурсах памяти	0,049	2	4	3	3	0,098	0,196	0,147	0,147
5. Функциональная мощность (предоставляемые возможности)	0,1325	3	2	5	5	0,3975	0,265	0,6625	0,6625
6. Простота эксплуатации	0,049	5	4	3	4	0,245	0,196	0,147	0,196
7. Качество интеллектуального интерфейса	0,063	4	1	2	3	0,252	0,063	0,126	0,189
Экономические критерии оценки эффективности									
1. Конкурентоспособность продукта	0,07	1	2	4	5	0,07	0,14	0,28	0,35
2. Уровень проникновения на рынок	0,033	1	3	4	4	0,033	0,099	0,132	0,132
3. Цена	0,075	5	5	3	4	0,375	0,375	0,225	0,3
4. Послепродажное обслуживание	0,072	1	2	4	5	0,072	0,144	0,288	0,36
5. Срок выхода на рынок	0,083	1	4	3	3	0,083	0,332	0,249	0,249
Итого	1	30	37	43	50	2,5	3,0535	3,7505	4,33

Из рассчитанных показателей в таблице 3.1 можно сделать вывод о том, что конкурент К₃ имеет наиболее сильную конкурентоспособность по сравнению с аналогами, в то время как оцениваемая разработка уступает по показателю конкурентоспособности своим аналогам.

3.1.3 FAST-анализ

Стадия 1. Выбор объекта анализа

В качестве объекта анализа выступает объект исследования – программное решение для генерации тестовых данных.

Стадия 2. Описание главной, основных и вспомогательных функций, выполняемых объектом

1) Главная функция объекта – генерация тестовых данных для БД.

2) Основные функции:

- Соединение с целевой БД.
- Вывод результатов в удобном для пользователя виде.
- Предоставление возможности управлять входными параметрами (настройками генерации) в удобном для пользователя виде.

3) Вспомогательные функции:

- Хранение настроек пользователя.
- Соблюдение ограничений целостности целевой БД.
- Предоставление интерфейса для управления автоназначением генераторов.
- Возможность генерации для View (представлений).

Информация, полученная при выполнении данной стадии, представлена в таблице 3.2.

Таблица 3.2 - Классификация функций, выполняемых объектом исследования

Наименование объекта	Выполняемая функция	Ранг функции		
		Главная	Основная	Вспомогательная
Главное окно программы	Управление процессом генерации тестовых данных	X		
Окно «Подключение»	Соединение с БД		X	
Окно «Предварительный просмотр»	Вывод результатов		X	
Окно «Настройки генерации»	Управление входными параметрами		X	
Промежуточная БД	Хранение настроек пользователя			X
Бизнес-логика приложения	Соблюдение ограничений целостности			X
Окно «Редактор правил автоназначения»	Управление автоназначением генераторов			X
Окно «Список объектов БД»	Генерация для View			X

Стадия 3. Определение значимости выполняемых функций объектом

Для оценки значимости функций используется метод расстановки приоритетов, предложенный Блумбергом В.А. и Глуценко В.Ф. В основу данного метода положено расчетно-экспертное определение значимости каждой функции.

Содержание матрицы смежности функций приведено в таблице 3.3.

Таблица 3.3 - Матрица смежности функций объекта

	Генерация тестовых данных	Соединение с БД	Вывод результатов	Управление входными параметрами	Хранение настроек	Ограничения целостности	Автоназначение генераторов	Генерация для View
Генерация тестовых данных	=	=	>	>	>	>	>	>
Соединение с БД	=	=	>	>	>	>	>	>
Вывод результатов	<	<	=	<	>	<	>	<
Управление входными параметрами	<	<	>	=	>	=	>	>

Хранение настроек	<	<	<	<	=	<	<	<
Ограничения целостности	<	<	>	=	>	=	>	>
Автоназначение генераторов	<	<	<	<	>	<	=	>
Генерация для View	<	<	>	<	>	<	<	=

Далее приведена преобразованная матрица – матрица количественных соотношений функций (таблица 3.4).

Таблица 3.4 - Матрица количественных соотношений функций

	Генерация тестовых данных	Соединение с БД	Вывод результатов	Управление входными параметрами	Хранение настроек	Ограничения целостности	Автоназначение генераторов	Генерация для View	ИТОГО
Генерация тестовых данных	1	1	1,5	1,5	1,5	1,5	1,5	1,5	11
Соединение с БД	1	1	1,5	1,5	1,5	1,5	1,5	1,5	11
Вывод результатов	0,5	0,5	1	0,5	1,5	0,5	1,5	0,5	6,5
Управление входными параметрами	0,5	0,5	1,5	1	1,5	1	1,5	1,5	9
Хранение настроек	0,5	0,5	0,5	0,5	1	0,5	0,5	0,5	4,5
Ограничения целостности	0,5	0,5	1,5	1	1,5	1	1,5	1,5	9
Автоназначение генераторов	0,5	0,5	0,5	0,5	1,5	0,5	1	1,5	6,5
Генерация для View	0,5	0,5	1,5	0,5	1,5	0,5	0,5	1	6,5
									$\Sigma=64$

Определение относительной значимости функций:

Генерация тестовых данных $11/64 = 0,171875$; соединение с БД $11/64 = 0,171875$; вывод результатов $6,5/64 = 0,101563$; управление входными параметрами $9/64 = 0,140625$; хранение настроек $4,5/64 = 0,070313$; ограничения целостности $9/64 = 0,140625$; автоназначение генераторов $6,5/64 = 0,101563$; генерация для View $6,5/64 = 0,101563$.

Стадия 4. Анализ стоимости функций, выполняемых объектом исследования

Для расчёта были взяты следующие исходные данные: средняя заработная плата программиста в г. Томск, равная 281,25 руб. в час [13], стоимость лицензии инструментального ПО MS Visual Studio 2017 Community – бесплатно для организаций с числом разработчиков до 5 человек [14].

Расчет стоимости функций приведен в таблице 3.5.

Таблица 3.5 - Определение стоимости функций, выполняемых объектом исследования

Наименование объекта	Выполняемая функция	Затрачено, человеко-час	Стоимость инструментального ПО, руб.	Заработная плата программиста, руб.	Себестоимость, руб.
Главное окно программы	Управление процессом генерации тестовых данных	160	0	45000	45000
Окно «Подключение»	Соединение с БД	4	0	1125	1125
Окно «Предварительный просмотр»	Вывод результатов	8	0	2250	2250
Окно «Настройки генерации»	Управление входными параметрами	16	0	4500	4500
Промежуточная БД	Хранение настроек пользователя	8	0	2250	2250
Бизнес-логика приложения	Соблюдение ограничений целостности	40	0	11250	11250

Окно «Редактор правил автоназначения»	Управление автоназначением генераторов	20	0	5625	5625
Окно «Список объектов БД»	Генерация для View	16	0	4500	4500

Стадия 5. Построение функционально-стоимостной диаграммы объекта и её анализ.

На рис. 40 приведена в виде функционально-стоимостной диаграммы обобщенная информация, собранная в рамках предыдущих стадий.

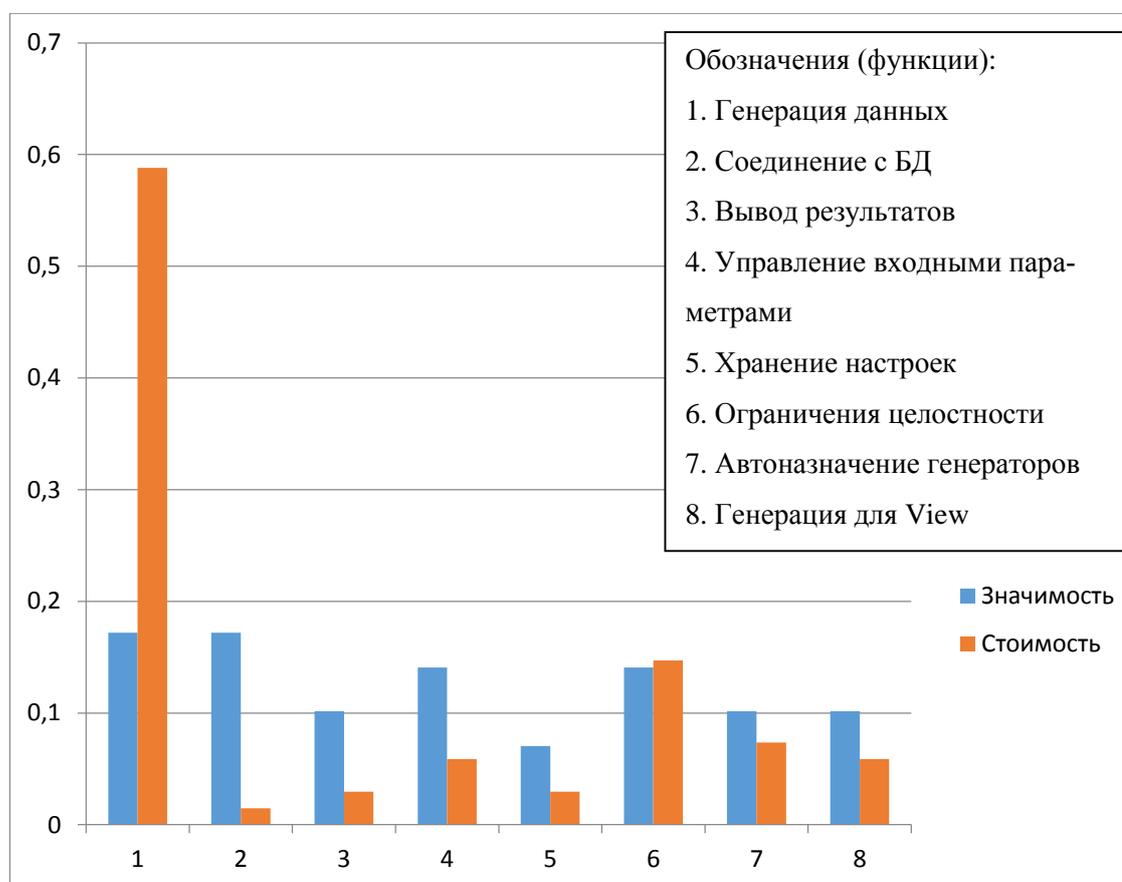


Рис. 40. Функционально-стоимостная диаграмма

Из явных диспропорций значимости и стоимости первой и второй пары столбцов в функционально-стоимостной диаграмме, напрашивается вывод, что усилия в ходе реализации проекта были распределены неправильно. Однако данное несоответствие объясняется спецификой проекта, а именно: значимость функций «Генерация данных» и «Соединение с БД» равная, т.к. без соединения с БД не может идти и речи о генерации данных, а без генерации

нет никакого смысла подключаться к БД. В то же время, задача соединения с БД является типовой и поэтому легко решается стандартными инструментами, отсюда и малое время, затраченное на её выполнение.

Кроме того, из диаграммы видно, что в функции 3,4 и 5 заложено недостаточно стоимости для обеспечения нужного уровня качества.

Стадия 6. Оптимизация функций, выполняемых объектом

Для повышения качества объекта исследования необходимо увеличение трудозатрат на доработку и оптимизацию функций «Вывод результатов», «Управление входными параметрами», «Хранение настроек».

3.1.4 Оценка готовности проекта к коммерциализации

В таблице 3.6 приведена форма, содержащая показатели о степени проработанности проекта с позиции коммерциализации и компетенции разработчика проекта.

Таблица 3.6 - Бланк оценки степени готовности проекта к коммерциализации

№ п/п	Наименование	Степень проработанности научного проекта	Уровень имеющихся знаний у разработчика
1.	Определен имеющийся научно-технический задел	3	3
2.	Определены перспективные направления коммерциализации научно-технического задела	2	2
3.	Определены отрасли и технологии (товары, услуги) для предложения на рынке	4	3
4.	Определена товарная форма научно-технического задела для представления на рынок	2	2
5.	Определены авторы и осуществлена охрана их прав	1	2
6.	Проведена оценка стоимости интеллектуальной собственности	2	1
7.	Проведены маркетинговые исследования рынков сбыта	1	1
8.	Разработан бизнес-план коммерциализации научной разработки	1	1
9.	Определены пути продвижения научной разработки на рынок	1	1
10.	Разработана стратегия (форма) реализации научной разработки	1	2

11.	Проработаны вопросы международного сотрудничества и выхода на зарубежный рынок	1	1
12.	Проработаны вопросы использования услуг инфраструктуры поддержки, получения льгот	1	1
13.	Проработаны вопросы финансирования коммерциализации научной разработки	1	1
14.	Имеется команда для коммерциализации научной разработки	1	2
15.	Проработан механизм реализации научного проекта	1	1
	ИТОГО БАЛЛОВ	23	24

Таким образом, можно сделать вывод об уровне перспективности «ниже среднего» для разработки и готовности разработчика к её коммерциализации.

3.2 Организация и планирование работ

В рамках планирования научного проекта был построен календарный план проекта в виде линейного графика в таблице (таблица 3.7) и его иллюстрация работы в виде диаграммы Ганта (таблица 3.8). Диаграммы Ганта строилась с разбивкой по месяцам и декадам который охватывает весь период времени выполнения научного проекта.

Таблица 3.7 - Календарный план проекта

Код	Название	Длительность, дни	Дата начала работ	Дата окончания работ	Состав участников
1	Постановка целей и задач	1	1.02	1.02	НР
2	Изучение предметной области, анализ аналогов	14	2.02	16.02	И
3	Формирование списка требований	7	17.02	24.02	И
4	Проектирование архитектуры	15	25.02	9.03	И
5	Проектирование служебной базы данных	12	10.03	22.03	И
6	Проектирование модулей генерации данных	20	23.03	13.04	И

7	Проектирование бизнес-логики	21	14.04	4.05	И
8	Проектирование пользовательского интерфейса	15	5.05	20.05	И
9	Оформление пояснительной записки	14	21.05	05.06	И
10	Проверка работы, подведение итогов	4	05.06	09.06	НР, И
Итого:		123			
Инженер		118			
Научный руководитель		5			

Таблица 3.8 - Диаграмма Ганта

Код	Исполнители	Кол. дн.	Февраль			Март			Апрель			Май			Июнь		
			1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
1	НР	1	■														
2	И	14	■	■													
3	И	7		■													
4	И	15			■	■											
5	И	12				■											
6	И	20					■	■	■								
7	И	21							■	■	■						
8	И	15									■	■					
9	И	14											■	■	■		
10	НР, И	4													■	■	

3.3 Расчёт сметы затрат на выполнение проекта

3.3.1 Расчёт заработной платы

Данная статья расходов включает заработную плату научного руководителя и инженера, а также премии, входящие в фонд заработной платы. Расчет основной заработной платы выполняется на основе трудоёмкости выполнения каждого этапа и величины месячного оклада исполнителя.

Величина месячного оклада научного руководителя (МОНР) получена из открытых данных, размещенных на официальном сайте Национального исследовательского Томского политехнического университета [15]. Величина месячного оклада инженеров (МОИ) берется как месячный оклад инженера кафедры.

Основной расчет фонда заработной платы выполняется по формуле:

$$ЗП_{\text{дн-т}} = \text{МО}/N, \quad (3.1)$$

где МО – месячный оклад, руб.;

N – количество рабочих дней в месяц, при шестидневной рабочей неделе – $N = 24,91$, а при пятидневной рабочей неделе – $N = 20,58$.

Среднедневная заработная плата научного руководителя равна:

$$ЗП_{\text{дн-т}} = \frac{26\,300}{24,91} = 1\,055,8 \frac{\text{руб.}}{\text{раб. день}}.$$

А среднедневная тарифная заработная плата инженеров равна

$$ЗП_{\text{дн-т}} = \frac{9800}{24,91} = 393,45 \frac{\text{руб.}}{\text{раб. день}}.$$

Затраты времени по каждому исполнителю в рабочих днях взяты из таблицы 5.4. Для перехода от тарифной суммы заработка исполнителя, связанной с участием в проекте, к соответствующему полному заработку необходимо будет тарифную сумму заработка исполнителя, связанной с участием в проекте умножить на интегральный коэффициент. Интегральный коэффициент находится по формуле:

$$K_{\text{и}} = K_{\text{пр}} \cdot K_{\text{доп.ЗП}} \cdot K_{\text{р}}, \quad (3.2)$$

где $K_{\text{пр}}$ – коэффициент премий, $K_{\text{пр}} = 1,1$;

$K_{\text{доп.ЗП}}$ – коэффициент дополнительной зарплаты, при шестидневной рабочей неделе $K_{\text{доп.ЗП}} = 1,188$, а при пятидневной рабочей неделе $K_{\text{доп.ЗП}} = 1,113$;

$K_{\text{р}}$ – коэффициент районной надбавки, $K_{\text{р}} = 1,3$.

Результаты вычислений представлены в таблице 3.9.

Таблица 3.9 – Затраты на заработную плату

Исполнитель	Оклад, руб./мес	ЗП _{дн-т} , руб./раб.день	Затраты вре- мени, раб.дни	Кoeffи- циент	Фонд з/платы, руб.
НР	26300	1055,8	5	1,699	8969,02
И	9800	393,45	118	1,699	78879,64
Итого:					87848,66

3.3.2 Материальные затраты

Материальные затраты – элемент себестоимости продукта. В ходе выполнения научно-исследовательской работы для разработки алгоритма не были приобретены услуги. Использованные программы и библиотеки бесплатны для учащихся в заведениях высшего образования и находятся в открытом доступе.

Перечень материалов и затраты на них приведены в таблице 3.10.

Таблица 3.10 - Затраты на материалы

Наименование	Кол-во	Ед. измере- ния	Цена за еди- ницу, руб.	Сумма, руб.
Печать на бу- маге А4	134	страницы, шт.	2,5	335
Всего за материалы				335
Транспортно-заготовительные расходы (3-5%)				13,4
Итого				348,4

3.3.3 Отчисления на социальные нужды

Ставка взноса в социальные фонды при выполнении научных работ установлен в размере 27,1% от заработной платы. Размер взноса рассчитываются по формуле:

$$C_{\text{соц}} = C_{\text{зп}} \cdot 0,271, \quad (3.3)$$

где $C_{\text{зп}}$ – размер заработной платы.

Подставив необходимые значения в формулу 5.10 получим:

$$C_{\text{соц}} = 87848,66 \cdot 0,271 = 23806,99 \text{ руб.}$$

3.3.4 Затраты на электроэнергию

Затраты на электроэнергию рассчитываются по формуле:

$$C_{\text{эл.об.}} = P_{\text{об}} \cdot t_{\text{об}} \cdot C_{\text{э}}, \quad (3.4)$$

где $P_{\text{об}}$ – мощность, потребляемая оборудованием, кВт;

$t_{\text{об}}$ – время работы оборудования, час;

$C_{\text{э}}$ – тариф на 1 кВт·час. Для ТПУ, $C_{\text{э}} = 5,8 \text{ руб./кВт} \cdot \text{час}$.

Время работы оборудования вычисляется на основе итоговых данных таблицы 3.7 для инженера ($T_{\text{рд}}$) из расчета, что продолжительность рабочего дня равна 8 часов.

$$t_{\text{об}} = T_{\text{рд}} \cdot K_t, \quad (3.5)$$

где K_t – коэффициент использования оборудования по времени, $K_t = 0,9$.

Мощность, потребляемая оборудованием, определяется по формуле:

$$P_{\text{об}} = P_{\text{ном}} \cdot K_C, \quad (3.6)$$

где K_C – коэффициент загрузки;

$P_{\text{ном}}$ – номинальная мощность оборудования, кВт. Для технологического оборудования малой мощности $K_C = 1$.

Результаты вычислений представлены в таблице 3.11.

Таблица 3.11 – Затраты на электроэнергию технологическую

Наименование оборудования	Время работы оборудования $t_{\text{об}}$, час	Потребляемая мощность $P_{\text{об}}$, кВт	Затраты $C_{\text{эл.об.}}$, руб.
Персональный компьютер инженера	944	0,45	424,8
Итого:			424,8

3.3.5 Накладные расходы

Накладные расходы составляют 40% от всей суммы затрат. Расчет накладных расходов ведется по следующей формуле:

$$C_{\text{нак}} = S_{\text{об.сум.зат}} \cdot 0,4, \quad (3.7)$$

где $S_{\text{об.сум.зат}}$ – общая сумма всех затрат, составляет 112428,85 рубля.

$$C_{\text{нак}} = 112428,85 \cdot 0,4 = 44971,54$$

3.3.6 Расчет общей себестоимости разработки

Проведя расчет сметы затрат на разработку, можно определить общую стоимость разработки проекта «Разработка и программная реализация алгоритма сегментации и распознавания автомобильных номеров» (таблица 3.12)

Таблица 3.12 - Смета затрат на разработку проекта

Статья затрат	Условное обозначение	Сумма, руб.
1. Заработная плата	$C_{\text{зп}}$	87848,66
2. Материальные затраты	$C_{\text{мат}}$	348,4
3. Отчисления на социальные нужды	$C_{\text{соц}}$	23806,99
4. Расходы на электроэнергию	$C_{\text{эл}}$	424,8
5. Накладные расходы	$C_{\text{нак}}$	44971,54
Итого:		157400,39

Таким образом, затраты на разработку составили $C = 157400,39$ руб.

3.4 Оценка научно-технического уровня проекта

В таблице 3.13 указано соответствие качественных уровней проекта значениям интегрального показателя научно-технического уровня работы.

Таблица 3.13 - Оценка научно-технического уровня проекта

Фактор НТУ	Значимость	Уровень фактора	Выбранный балл	Обоснование выбранного балла
Уровень новизны	0,4	Относительно новая	3	Разрабатывается, т.к. является бесплатной, в отличие от аналогов
Теоретический уровень	0,1	Разработка способа	6	Разработка программы
Возможность реализации	0,5	В течение первых лет	10	Быстрая разработка с помощью современных инструментальных средств

Интегральный показатель научно-технического уровня составляет:

$$I_{НТУ} = 0,4 \cdot 3 + 0,1 \cdot 6 + 0,5 \cdot 10 = 6,8.$$

Оценка научно-технического уровня проводится с помощью данных, указанных в таблице 3.14.

Таблица 3.14 - Оценка уровня научно-технического эффекта

Уровень НТЭ	Показатель НТЭ
Низкий	1–4
Средний	4–7
Высокий	8–10

Таким образом, научно-исследовательская работа имеет средний уровень научно-технического эффекта.

4. СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ

Обычно при разработке и последующем тестировании производительности, отказоустойчивости и корректности работы информационной системы возникает потребность заполнения её баз данных тестовыми данными.

Существует отдельный класс программных средств, предназначенных для решения задачи генерации тестовых данных.

Разрабатываемая программная система – генератор тестовых данных для баз данных под управлением SQL Server – относится к классу средств автоматизации тестирования. Цель применения данного программного средства: обеспечение повышения производительности путём автоматизации производства.

Область применения: автоматизация тестирования при разработке программного обеспечения.

Возможные пользователи: профессиональные разработчики и тестеры БД.

Данный раздел посвящен анализу вредных и опасных факторов производственной среды как для разработчиков, так и для пользователей.

4.1. Профессиональная социальная безопасность

4.1.1 Анализ вредных и опасных факторов на рабочем месте

Для обеспечения производственной безопасности необходимо проанализировать воздействия на человека вредных и опасных производственных факторов, которые могут возникать на рабочем месте.

Производственный фактор считается вредным, если воздействие этого фактора на человека может привести к его заболеванию. Производственный фактор считается опасным, если его воздействие может привести к травме [166].

Все производственные факторы классифицируются по группам элементов: физические, химические, биологические и психофизические. Для данной работы целесообразно рассмотреть физические и психофизические вредные и опасные факторы производства, характерные для рабочей зоны программиста, разработчика приложения, пользователя. Выявленные факторы представлены в таблице 4.1.

Таблица 4.1 - Вредные и опасные производственные факторы при выполнении работ за ПЭВМ

Источник фактора, наименование видов работ	Факторы (по ГОСТ 12.0.003-74)		Нормативные документы
	Вредные	Опасные	
1) Работа за персональным компьютером	1) Недостаток естественного света; 2) Недостаточная освещенность рабочей зоны; 3) Монотонность труда.	1) Опасность поражения электрическим током; 2) Опасность возникновения пожара.	1) СН 2.2.4/2.1.8.562-96; 2) СанПиН 2.2.4.548-96; 3) СанПиН 2.2.2/2.4.1340-03; 4) СП 52.13330.2011; 5) ГОСТ Р 12.1.019-2009 ССБТ; 6) СНиП 21-01-97.

4.1.1.1 Вредные производственные факторы

4.1.1.1.1 Недостаток естественного света

Недостаток естественного света относится к физическим опасным и вредным факторам. При длительном солнечном голодании снижаются умственные способности и работоспособность, очень быстро наступает утомление и раздражение, уменьшается подвижность, снижается иммунитет, организм плохо снабжается питательными веществами и кислородом и т.д. Учи-

тывая степень благотворного влияния естественного света на организм человека, гигиена труда требует максимального использования естественного освещения.

4.1.1.1.2 Недостаточная освещенность рабочей зоны

Недостаточная освещенность рабочей зоны является вредным производственным фактором, возникающим при работе с ПЭВМ, уровни которого регламентируются СП 52.13330.2011.

Причиной недостаточной освещенности являются недостаточность естественного освещения, недостаточность искусственного освещения, пониженная контрастность.

Работа с компьютером подразумевает постоянный зрительный контакт с дисплеем ПЭВМ и занимает от 80% рабочего времени. Недостаточность освещения снижает производительность труда, увеличивает утомляемость и количество допускаемых ошибок, а также может привести к появлению профессиональных болезней зрения.

Разряд зрительных работ программиста и оператора ПЭВМ относится к разряду III и подразряду Г (работы высокой точности). В таблице 4.2 представлены нормативные показатели искусственного освещения при работах заданной точности [17].

Таблица 4.2 – Требования к освещению помещений промышленных предприятий для операторов ПЭВМ

Характеристика зрительной работы	Наименьший или эквивалентный размер объекта различения, мм	Разряд зрительной работы	Подразряд зрительной работы	Контраст объекта с фоном	Характеристика фона	Искусственное освещение		
						Освещённость, лк		
						При системе комбинированного освещения		При системе общего освещения
всего	В том числе от общего							
Высокой точности	0,264	III	г	Средний, большой	Светлый, средний	400	200	200

Для создания и поддержания благоприятных условий освещения для операторов ПЭВМ, их рабочие места должны соответствовать санитарно-эпидемиологическим правилам СанПиН 2.2.2/2.4.1340-03. Для рассеивания естественного освещения следует использовать жалюзи на окнах рабочих помещений. В качестве источников искусственного освещения должны быть использованы люминесцентные лампы, лампы накаливания – для местного освещения [18].

4.1.1.1.3 Монотонность труда

При работе с ПЭВМ основным фактором, влияющим на нервную систему программиста или пользователя, является огромное количество информации, которое он должен воспринимать. Это является сложной задачей, которая очень сильно влияет на сознание и психофизическое состояние из-за монотонности работы. Поэтому меры, позволяющие снизить воздействие этого вредного производственного фактора, которые регулируются СанПиН 2.2.2/2.4.1340-03, являются важными в работе оператора ПЭВМ. Они позволяют увеличить производительность труда и предотвратить появление профессиональных болезней.

4.1.1.2 Опасные производственные факторы

4.1.1.2.1 Опасность поражения электрическим током

Поражение электрическим током является опасным производственным фактором и, поскольку программист имеет дело с электрооборудованием, то вопросам электробезопасности на его рабочем месте должно уделяться особое внимание. Нормы электробезопасности на рабочем месте регламентируются СанПиН 2.2.2/2.4.1340-03, вопросы требований к защите от поражения электрическим током освещены в ГОСТ Р 12.1.019-2009 ССБТ.

Электробезопасность – система организационных и технических мероприятий и средств, обеспечивающих защиту людей от вредного и опасного воздействия электрического тока, электрической дуги, электромагнитного поля и статического электричества.

Опасность поражения электрическим током усугубляется тем, что человек не в состоянии без специальных приборов обнаружить напряжение дистанционно.

4.1.1.2.2 Опасность возникновения пожара

Возникновение пожара является опасным производственным фактором, т.к. пожар на предприятии наносит большой материальный ущерб, а также часто сопровождается травмами и несчастными случаями. Регулирование пожаробезопасности производится СНиП 21-01-97.

В помещениях с ПЭВМ повышен риск возникновения пожара из-за присутствия множества факторов: наличие большого количества электронных схем, устройств электропитания, устройств кондиционирования воздуха; возможные неисправности электрооборудования, освещения, или неправильная их эксплуатация может послужить причиной пожара.

Возможные виды источников воспламенения:

- Искра при разряде статического электричества;
- Искры от электрооборудования;

- Искры от удара и трения;
- Открытое пламя [19].

4.1.2 Обоснование мероприятий по защите от действия вредных и опасных факторов

4.1.2.1 Мероприятия по улучшению условий труда по фактору световая среда

В перечень нормируемых параметров световой среды входят: КЕО, % - коэффициент естественной освещенности; Е, лк - освещенность рабочей поверхности; прямая блескость (показатель ослепленности); Кп, % - коэффициент пульсации освещенности; отраженная блескость; L, кд/м² - яркость; С, отн. ед. – неравномерность распределения яркости в поле зрения пользователя компьютером. Нормируемые показатели фактора «световая среда» приведены в таблице 4.3.

Таблица 4.3 - Перечень нормируемых показателей фактора «световая среда»

Показатель	Единица измерений
Коэффициент естественной освещенности	%
Освещенность(Е)	лк
Коэффициент пульсации освещенности (Кп)	%
Яркость (L)	кд/м ²
Неравномерность распределения яркости (С)	-
Прямая блескость	-
Отраженная блескость	-
Визуальные параметры ВДТ	-

Методика проведения измерений параметров световой среды устанавливается в соответствии с:

- Методическими указаниями МУК 4.3.2812-10 «Инструментальный контроль и оценка освещения рабочих мест»;
- Методическими указаниями МУ 2.2.4.706-98/МУ ОТ РМ 01-98 «Оценка освещения рабочих мест».

Оценка условий труда по фактору «световая среда» производится в соответствии со следующими документами:

- Свод правил СП 52.13330.2011 «Естественное и искусственное освещение. Актуализированная редакция СНиП 23-05-95»;
- Санитарные правила и нормы СанПиН 2.2.1/2.1.1.1278-03 «Гигиенические требования к естественному, искусственному и совмещенному освещению жилых и общественных зданий»;
- Санитарные правила и нормы СанПиН 2.2.2/2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы»;
- Отраслевые нормативные и методические документы.

4.1.2.2 Мероприятия по улучшению условий работы в монотонном режиме

Организация работы с ПЭВМ осуществляется в зависимости от вида и категории трудовой деятельности. Виды трудовой деятельности разделяются на 3 группы: группа А – работа по считыванию информации с экрана с предварительным запросом; группа Б – работа по вводу информации; группа В – творческая работа в режиме диалога с ПЭВМ. Работа программиста-разработчика рассматриваемой в данной работе относится к группам А и Б, в то время, как деятельность пользователя приложения относится к группе В. Категории трудовой деятельности, различаются по степени тяжести выполняемых работ. Для снижения воздействия рассматриваемого вредного фактора предусмотрены регламентированные перерывы для каждой группы работ [20] – таблица 4.4.

Таблица 4.4 - Суммарное время регламентированных перерывов в зависимости от продолжительности работы, вида категории трудовой деятельности с ПЭВМ

Категория работы с ПЭВМ	Уровень нагрузки за рабочую смену при видах работ с ПЭВМ			Суммарное время регламентированных перерывов, мин.	
	группа А, количество знаков	группа Б, количество знаков	группа В, ч	при 8-часовой смене	при 12-часовой смене
I	до 20 000	до 15 000	до 2	50	80
II	до 40 000	до 30 000	до 4	70	110
III	до 60 000	до 40 000	до 6	90	140

Для предупреждения преждевременной утомляемости пользователей ПЭВМ рекомендуется организовывать рабочую смену путем чередования работ с использованием ПЭВМ и без него. В случаях, когда характер работы требует постоянного взаимодействия с компьютером (работа программиста-разработчика) с напряжением внимания и сосредоточенности, при исключении возможности периодического переключения на другие виды трудовой деятельности, не связанные с ПЭВМ, рекомендуется организация перерывов на 10–15 мин. через каждые 45–60 мин. работы. При высоком уровне напряженности работы рекомендуется психологическая разгрузка в специально оборудованных помещениях [18].

4.1.2.3 Мероприятия по предотвращению возможности поражения электрическим током

Основным организационным мероприятием по обеспечению безопасности является инструктаж и обучение безопасным методам труда, а также проверка знаний правил безопасности и инструкций в соответствии с занимаемой должностью применительно к выполняемой работе.

К мероприятиям по предотвращению возможности поражения электрическим током относятся:

- С целью защиты от поражения электрическим током, возникающим между корпусом приборов и инструментом при пробое сетевого

напряжения на корпус, корпуса приборов и инструментов должны быть заземлены;

- При включенном сетевом напряжении работы на задней панели корпуса приборов должны быть запрещены;
- Все работы по устранению неисправностей должен производить квалифицированный персонал;
- Необходимо постоянно следить за исправностью электропроводки [18].

4.1.2.4 Мероприятия по профилактике пожаров

Для профилактики организации действий при пожаре должен проводиться следующий комплекс организационных мер: должны обеспечиваться регулярные проверки пожарной сигнализации, первичных средств пожаротушения; должен проводиться инструктаж и тренировки по действиям в случае пожара; не должны загромождаться или блокироваться пожарные выходы; должны выполняться правила техники безопасности и технической эксплуатации электроустановок; во всех служебных помещениях должны быть установлены «Планы эвакуации людей при пожаре и других ЧС», регламентирующие действия персонала при возникновении пожара.

Для предотвращения пожара помещение с ПЭВМ должно быть оборудовано первичными средствами пожаротушения: углекислотными огнетушителями типа ОУ-2 или ОУ-5; пожарной сигнализацией, а также, в некоторых случаях, автоматической установкой объемного газового пожаротушения [20].

4.2. Экологическая безопасность

4.2.1 Анализ влияния объекта исследования на окружающую среду

В данном разделе рассматривается воздействие проекта на окружающую среду.

В ходе выполнения ВКР и дальнейшем использовании отсутствуют выбросы каких-либо вредных веществ в атмосферу и гидросферу, следовательно, загрязнение воздуха и воды не происходит.

Люминесцентные лампы, применяющиеся для искусственного освещения рабочих мест, также требуют особой утилизации, т.к. в них присутствует от 10 до 70 мг ртути, которая относится к чрезвычайно-опасным химическим веществам и может стать причиной отравления живых существ, а также загрязнения атмосферы, гидросферы и литосферы. Сроки службы таких ламп составляют около 5-ти лет, после чего их необходимо сдавать на переработку в специальных пунктах приема.

Во время эксплуатации программных продуктов могут образовываться отходы в виде неисправных или устаревших комплектующих персональных компьютеров.

4.2.2 Обоснование мероприятий по защите окружающей среды

Для уменьшения вредного влияния на литосферу необходимо производить сортировку отходов и обращаться в службы по утилизации для дальнейшей переработки или захоронения. [21]

В основном, организации, занимающиеся приёмом и утилизацией ртути содержащих отходов, принимают люминесцентные лампы в массовых количествах. Лампа состоит из электронного блока — выгодный компонент для реставрации и утилизации; колба и цоколь также ценное сырье.

Такие лампы нельзя выкидывать в мусоропровод или уличные контейнеры, а нужно отнести в районную дирекцию единичного заказчика (ДЕЗ) или ремонтно-эксплуатационное управление (РЭУ), где есть специальные контейнеры. Там они принимаются бесплатно, основанием должна служить утилизация в соответствии с Управлением Федеральной службы по надзору в сфере защиты прав потребителей и благополучия человека по Томской области. Адреса пунктов приёма отработавших свой срок люминесцентных ламп можно найти в интернете [22].

4.3 Безопасность в чрезвычайных ситуациях

4.3.1 Анализ вероятных ЧС, которые могут возникнуть на рабочем месте

Чрезвычайные ситуации бывают техногенного, природного, биологического, социального или экологического характера.

При работе в кабинете могут возникнуть следующие классификации чрезвычайных ситуаций:

- Преднамеренные/непреднамеренные;
- Техногенные: взрывы, пожары, обрушение помещений, аварии на системах жизнеобеспечения/природные – связанные с проявлением стихийных сил природы.
- Экологические – это аномальные изменения состояния природной среды, такие как загрязнения биосферы, разрушение озонового слоя, кислотные дожди/ антропогенные – являются следствием ошибочных действий людей.
- Биологические – различные эпидемии, эпизоотии, эпифитотии;
- Социальные – это обстановка на определенной территории, сложившаяся в результате опасного социального явления, которое повлекло в результате человеческие жертвы, ущерб здоровью, имуществу или окружающей среды;
- Комбинированные.

Наиболее вероятная чрезвычайная ситуация, которая может возникнуть при работе с ПЭВМ – пожар, так как в современных ЭВМ очень высокая плотность размещения элементов электронных схем. В непосредственной близости друг от друга располагаются соединительные провода и кабели, при протекании по ним электрического тока выделяется значительное количество тепла, при этом возможно оплавление изоляции и возникновение возгорания.

4.3.2 Обоснование мероприятий по предотвращению ЧС и разработка порядка действия в случае возникновения ЧС

Меры, соблюдение которых поможет исключить с большой вероятностью возможность возникновения пожара:

- Для понижения воспламеняемости и способности распространять пламя кабели покрывают огнезащитным покрытием;
- При ремонтно-профилактических работах строго соблюдаются правила пожарной безопасности;
- Помещения, в которых должны располагаться ПЭВМ проектируют I или II степени огнестойкости;
- Каждое из помещений, где производится эксплуатация устройств ПЭВМ, должно быть оборудовано первичными средствами пожаротушения и обеспечено инструкциями по их применению. В качестве средств пожаротушения разрешается использование углекислотного огнетушителя типа ОУ-2, ОУ-5, а также порошковый тип. Применение пенных огнетушителей не допускается, так как жидкость пропускает ток;
- Устройства ПЭВМ необходимо устанавливать вдали от отопительных и нагревательных приборов (расстояние не менее 1 метра, в местах, где не затруднена их вентиляция и нет прямых солнечных лучей);
- Разрабатываются организационные меры по обучению персонала навыкам ликвидации пожара имеющимися в наличии средствами тушения пожара до прибытия пожарного подразделения [23].

При пожаре люди должны покинуть помещение в течение минимально возможного отрезка времени.

В помещениях с компьютерной техникой, недопустимо применение воды и пены ввиду опасности повреждения или полного выхода из строя дорогостоящего электронного оборудования.

Для тушения пожаров необходимо применять углекислотные и порошковые огнетушители, которые обладают высокой скоростью тушения, большим временем действия, возможностью тушения электроустановок, высокой эффективностью борьбы с огнем. Воду разрешено применять только во вспомогательных помещениях [20].

4.4 Правовые и организационные вопросы обеспечения безопасности

4.4.1 Специальные правовые нормы трудового законодательства

Если пользователь постоянно загружен работой с ЭВМ, приемлемой является поза сидя. В положении сидя основная нагрузка падает на мышцы, поддерживающие позвоночный столб и голову. В связи с этим при длительном сидении время от времени необходимо сменять фиксированные рабочие позы.

Исходя из общих принципов организации рабочего места, в нормативно-методических документах сформулированы требования к конструкции рабочего места.

Основными элементами рабочего места программиста являются: рабочий стол, рабочий стул (кресло), дисплей, клавиатура, мышь; вспомогательными - пюпитр, подставка для ног [24].

Взаимное расположение элементов рабочего места должно обеспечивать возможность осуществления всех необходимых движений и перемещений для эксплуатации и технического обслуживания оборудования [25].

Рабочие места с ЭВМ должны располагаться на расстоянии не менее 1,5 м от стены с оконными проемами, от других стен – на расстоянии 1 м, между собой – на расстоянии не менее 1,5 м. При размещении рабочих мест необходимо исключить возможность прямой засветки экрана источником естественного освещения.

При размещении ЭВМ на рабочем месте должно обеспечиваться пространство для пользователя величиной не менее 850 мм³. Для стоп должно быть предусмотрено пространство по глубине и высоте не менее 150 мм, по

ширине – не менее 530 мм. Располагать ЭВМ на рабочем месте необходимо так, чтобы поверхность экрана находилась на расстоянии 400 – 700 мм от глаз пользователя. Конструкция рабочего места и взаимное расположение всех его элементов (сиденье, органы управления, средства отображения информации и т.д.) должны соответствовать антропометрическим, физиологическим и психологическим требованиям, а также характеру работы [25].

Рабочее кресло обеспечивает поддержание рабочей позы в положении сидя, и чем длительнее это положение в течение рабочего дня, тем жестче должны быть требования к созданию удобных и правильных рабочих сидений.

Высота поверхности сиденья должна регулироваться в пределах 400 – 550 мм. Ширина и глубина его поверхности должна быть не менее 400 мм. Поверхность сиденья должна быть плоской, передний край – закругленным. Сиденье и спинка кресла должны быть полумягкими, с нескользящим, не электризующимся и воздухопроницаемым покрытием, материал которого обеспечивает возможность легкой очистки от загрязнения.

Опорная поверхность спинки стула должна иметь высоту 280 – 320 мм, ширину – не менее 380 мм и радиус кривизны горизонтальной плоскости – 400 мм. Расстояние спинки от переднего края сиденья должно регулироваться в пределах 260 – 400 мм.

Рабочее место должно быть оборудовано устойчивой и просто регулируемой подставкой для ног, располагающейся, по возможности, по всей ширине отводимого участка для ног. Подставка должна иметь ширину не менее 300 мм, глубину не менее 400 мм, регулировку по высоте до 150 мм и по углу наклона опорной поверхности подставки до 20. Поверхность подставки должна быть рифленой, по переднему краю иметь бортик высотой 10 мм.

При организации рабочего пространства необходимо учитывать индивидуальные антропометрические параметры пользователя с соответствующими допусками на возможные изменения рабочих поз и потребность в перемещениях.

Рациональной рабочей позой может считаться такое расположение тела, при котором ступни работника расположены на плоскости пола или на подставке для ног, бедра сориентированы в горизонтальной плоскости, верхние части рук – вертикальный угол локтевого сустава колеблется в пределах 70 – 90, запястья согнуты под углом не более чем 20, наклон головы – в пределах 15 – 20, а также исключены частые ее повороты [24].

4.4.2 Влияние разработки на организацию производственных процессов

Основной целью разработки является автоматизация процесса тестирования программного обеспечения, в частности информационных систем (ИС).

Автоматизация процесса тестирования позволяет ускорить процесс заполнения БД тестовыми данными, что благоприятным образом отразится на труде программиста или тестировщика. Сокращение времени, затрачиваемого на заполнения БД системы тестовыми данными, снижает влияние вредных факторов труда на ПЭВМ (монотонность работы) и, как следствие, повышает качество разрабатываемых систем.

С другой стороны, полагать что автоматизированная генерация тестовых данных способна полностью освободить от данной работы разработчика или тестировщика будет ошибочно.

Несмотря на все возможности гибкой настройки стартовых параметров, нельзя дать стопроцентную гарантию, что данное программное решение позволит генерировать полностью релевантные данные, отвечающие нуждам заказчика. Зачастую разработчику будет необходимо проводить дополнительный контроль входных параметров, для обеспечения уверенности, что генерируемые тестовые данные позволяют правильным образом продемонстрировать возможности разрабатываемой ИС.

ЗАКЛЮЧЕНИЕ

В результате выполнения работы были решены следующие основные задачи:

- проведены изучение и сравнительный анализ аналогов;
- выработаны требования к проектируемой системе;
- спроектирован графический интерфейс пользователя;
- реализован режим генерации данных для представлений;
- реализован модуль автоматизированного подбора стартовых параметров, добавлена возможность его гибкой настройки пользователем;
- реализована логика по обеспечению ограничений целостности первичного и внешних ключей;
- реализован модуль для задания пользователем собственных типов генераторов данных.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Databene Venerator — генерация тестовых данных [Электронный ресурс]. – Режим доступа: URL:<https://habrahabr.ru/post/169713/>, свободный (05.03.2017). – Загл. с экрана.
- 2 Релевантность – Викисловарь [Электронный ресурс]. – Режим доступа: URL: <https://ru.wiktionary.org/wiki/релевантность>, свободный (11.04.2017). – Загл. с экрана.
- 3 Понимание SQL. Глава 21. Изменение значений с помощью представлений [Электронный ресурс] – Режим доступа: URL: http://www.sql.ru/docs/sql/u_sql/ch21.shtml, свободный (12.05.2016). – Загл. с экрана.
- 4 Ограничения целостности реляционной модели данных [Электронный ресурс]. – Режим доступа: URL: http://www.mstu.edu.ru/study/materials/zelenkov/ch_4_3.html, свободный (11.04.2017). – Загл. с экрана.
- 5 Путеводитель по бесплатным инструментам для SQL Server - № 11, 2009 | Windows IT Pro | Издательство «Открытые системы» [Электронный ресурс]. – Режим доступа: URL: <http://www.osp.ru/win2000/2009/11/13000862/>, свободный (19.01.2016). – Загл. с экрана.
- 6 DataGenerator for SQLServer. Руководство пользователя – [Электронный ресурс] Режим доступа: URL: http://download2.sqlmanager.net/download/mssqlutils/doc/msdatagen_russian.pdf, свободный (19.01.2016). – Загл. с экрана.
- 7 SyBooks Online - SAP Sybase Power Designer 16.5 SP02 –“Generating Test Data to a Database” [Электронный ресурс]. – Режим доступа: URL:<http://infocenter.sybase.com/help/index.jsp?topic=/com.sybase.infocenter.dc38058.1652/doc/html/rad1232021421990.html>, свободный (19.01.2016). – Загл. с экрана.
- 8 Redgate Software “Add to cart - options for purchasing SQL Data Generator” [Электронный ресурс]. – Режим доступа: URL:

<https://www.red-gate.com/dynamic/purchase/product/sqldatagenerator>, свободный (19.01.2016). – Загл. с экрана.

- 9 Возможности dbForge Studio for SQL Server | devart [Электронный ресурс]. - Режим доступа: URL: <https://www.devart.com/ru/dbforge/sql/studio/features.html>, свободный (25.03.16). – Загл. с экрана.
- 10 Windows Forms–Википедия [Электронный ресурс]. – Режим доступа: URL:https://ru.wikipedia.org/wiki/Windows_Forms, свободный (25.10.2016). – Загл. с экрана.
- 11 SQLite – Википедия [Электронный ресурс]. – Режим доступа: URL: <https://ru.wikipedia.org/wiki/SQLite>, свободный (11.04.2017). – Загл. с экрана.
- 12 Обзор рынка программного обеспечения [Электронный ресурс]. Режим доступа: URL: http://www.marketing.spb.ru/mr/it/market_software.htm#4.3, (25.04.16) – Загл. с экрана.
- 13 Средняя зарплата программиста в Томске, статистика [Электронный ресурс]. Режим доступа: URL: <http://tomsk.trud.com/salary/909/3320.html>, (25.04.16) – Загл. с экрана.
- 14 Бесплатные инструменты для разработчиков, Visual Studio Community [Электронный ресурс]. Режим доступа: URL: <https://www.visualstudio.com/ru/vs/community/>, (05.05.16) – Загл. с экрана.
- 15 Должностные оклады ППС и педагогических работников с 01.06.2016 г. [Текст]: приказ ректора ТПУ от 25 мая 2016 г. №5994
- 16 Охрана труда. Основы безопасности жизнедеятельности // www.Grandars.ru. URL: <http://www.grandars.ru/shkola/bezopasnost-zhiznedeyatelnosti/ohrana-truda.html> (дата обращения: 11.03.2017).

- 17 Попов В.М. Психология безопасности профессиональной деятельности: учебное пособие / В. М. Попов; Новосибирский государственный технический университет. – Новосибирск: Изд-во Новосибирского государственного технического университета, 1996 г. – 155 с.
- 18 СанПиН 2.2.2/2.4.1340-03. Санитарно-эпидемиологические правила и нормы. Гигиенические требования к персональным электронно-вычислительным машинам и организации работы // Электронный фонд правовой и нормативно-технической документации. [Электронный ресурс] URL: <http://docs.cntd.ru/document/901865498> (10.03.2017) – Загл. с экрана.
- 19 ГОСТ Р 12.1.019-2009 ССБТ. Электробезопасность. Общие требования и номенклатура видов защиты // Электронный фонд правовой и нормативно-технической документации. [Электронный ресурс] URL: <http://docs.cntd.ru/document/1200080203> (11.03.2017) – Загл. с экрана.
- 20 Чрезвычайные ситуации при работе с ПЭВМ [Электронный ресурс] URL: http://studopedia.ru/8_107307_osveshchenie-pomeshcheniy-vichislitelnih-tsentrov.html (10.03.2017) Загл. – с экрана.
- 21 Постановление Правительства РФ от 03.09.2010 N 681 (ред. от 01.10.2013) "Об утверждении Правил обращения с отходами производства и потребления в части осветительных устройств, электрических ламп, ненадлежащие сбор, накопление, использование, обезвреживание, транспортирование и размещение которых может повлечь причинение вреда жизни, здоровью граждан, вреда животным, растениям и окружающей среде // Консультант Плюс. 2015. [Электронный ресурс] URL: http://www.consultant.ru/document/cons_doc_LAW_104420/e1b31c36ed1083efeb6cd9c63ed12f99e2ca77ed/#dst100007 (03.04.2017) – Загл. с экрана.

- 22 Как утилизировать люминесцентную лампу? [Электронный ресурс]
URL: <http://есо63.ru/lampalum.html> (03.04.2017) – Загл. с экрана.
- 23 Долин П.А. Справочник по технике безопасности. М.: Энергоатомиздат, 1984 г. – 824 с.
- 24 ГОСТ Р 50923-96 Дисплеи. Рабочее место оператора. Общие эргономические требования и требования к производственной среде. Методы измерения // Электронный фонд правовой и нормативно-технической документации. [Электронный ресурс] URL: <http://docs.cntd.ru/document/1200025975> (11.03.2017) – Загл. с экрана.
- 25 ГОСТ 22269-76 Система "Человек-машина". Рабочее место оператора. Взаимное расположение элементов рабочего места. Общие эргономические требования // Электронный фонд правовой и нормативно-технической документации. [Электронный ресурс] URL: <http://docs.cntd.ru/document/1200012834> (11.03.2017) – Загл. с экрана.

ПРИЛОЖЕНИЕ А

Раздел 2.5
General algorithms of operation

Раздел 2.6
The service database

Раздел 2.7
Compliance with the integrity constraints of the target database

Студент:

Группа	ФИО	Подпись	Дата
8ИМ5А	Мочилин Кирилл Алексеевич		

Консультант кафедры ИСТ:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент каф. ИСТ	Мирошниченко Е.А.	к.т.н.		

Консультант – лингвист кафедры ИЯИК:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель каф. ИЯИК	Горбатова Т.Н.			

2.5 General algorithms of operation

General algorithm of the program operation via Unified Modelling Language (UML) notation is shown on fig.14.

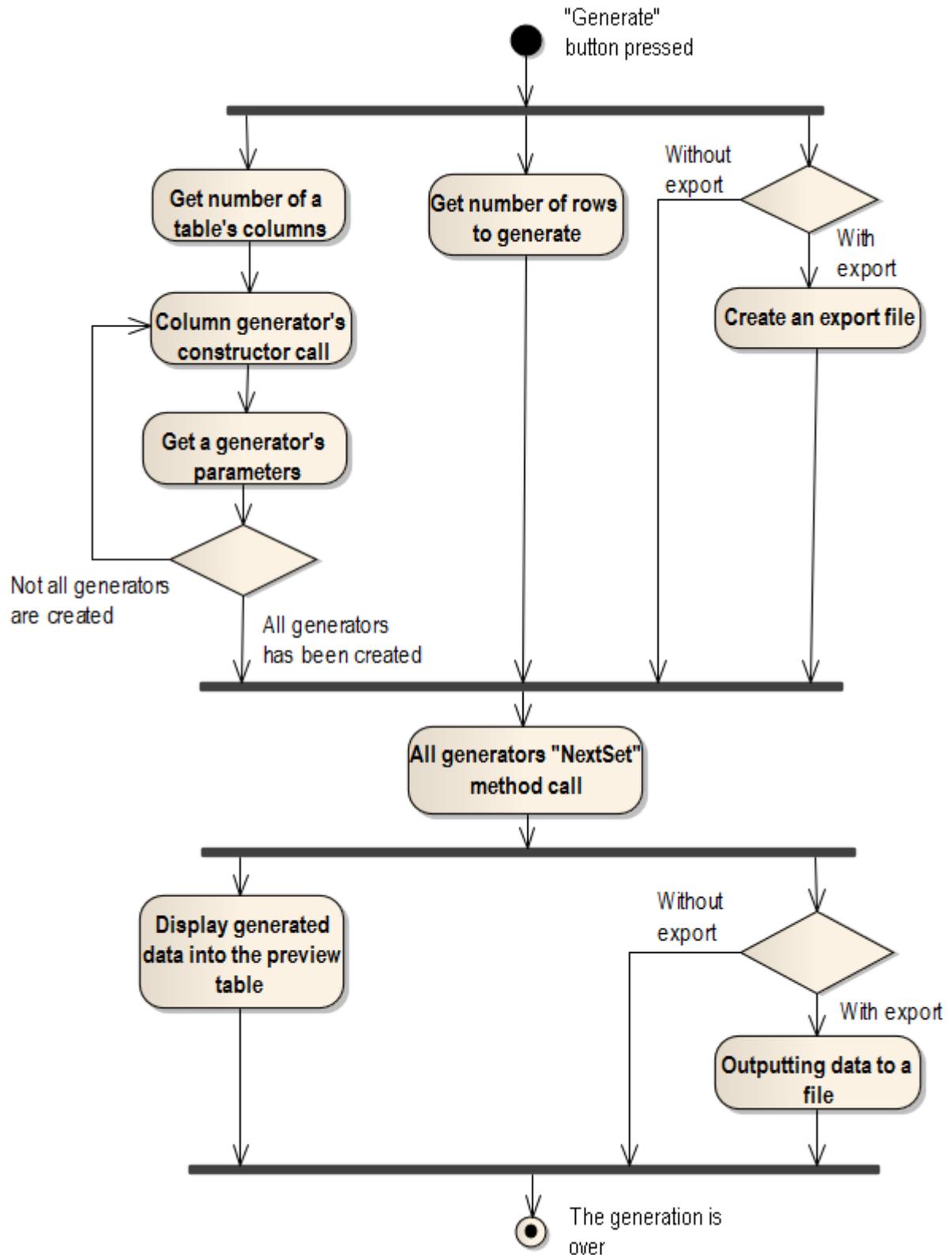


Fig.14. General algorithm of the program operation

An opportunity to export generated values into a SQL-file is available as an optional feature of a user's interface. In this case, a text file with the extension «sql» will be created in the subdirectory «Export» of the program's root directory. The following text is added into this file after a generation of every new data cortege:

```
«INSERT INTO [database name].[user name].[table name]
([values list])
VALUES
('added values')»
```

The algorithm of a text data generator NextSet method via UML activity diagram is shown on fig.15.

This algorithm is based on a sample of random values from a predetermined set of words.

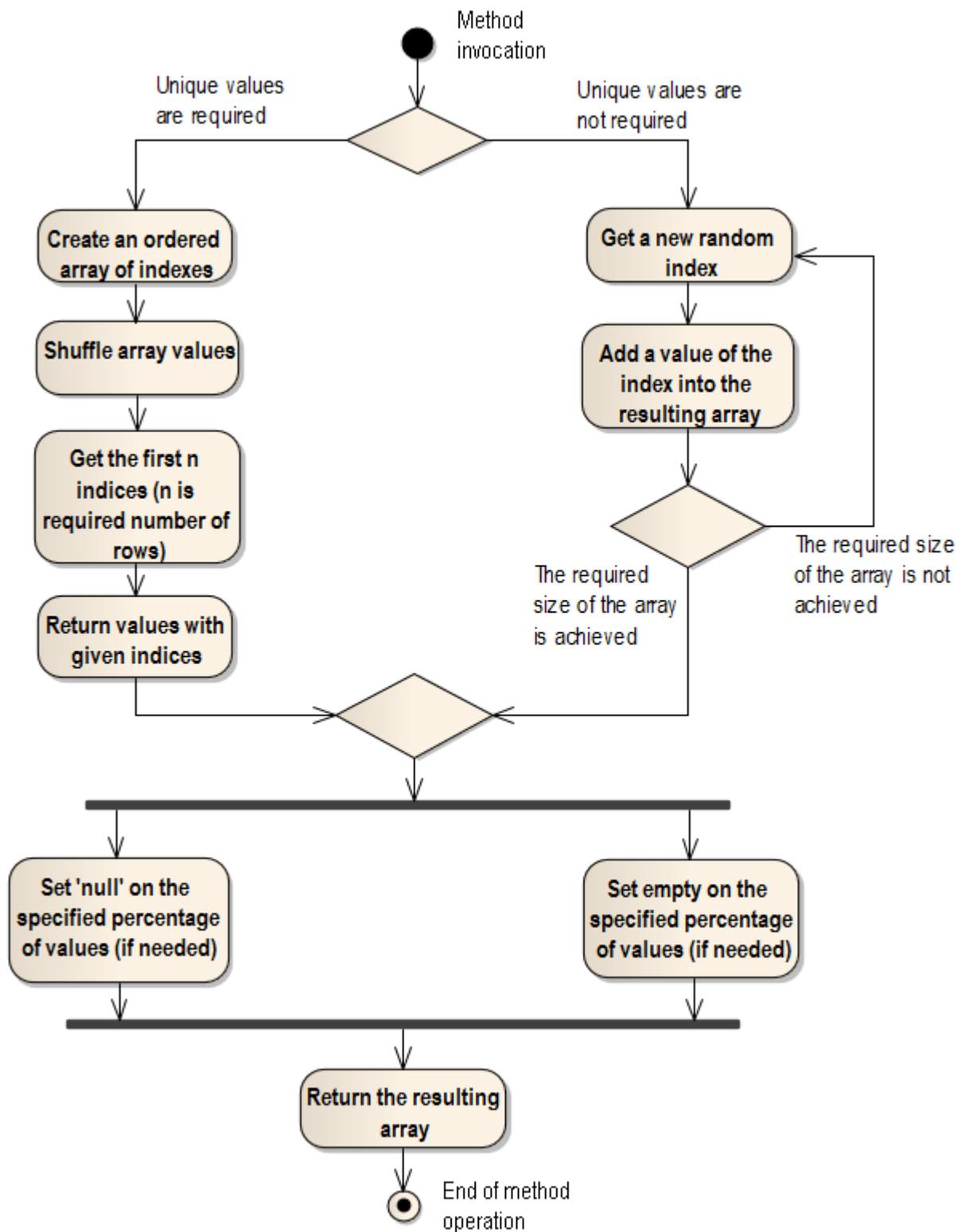


Fig.15. The algorithm of a text data generator operation

The algorithm of the NextSet method integer data generator via UML activity diagram is shown on fig.16.

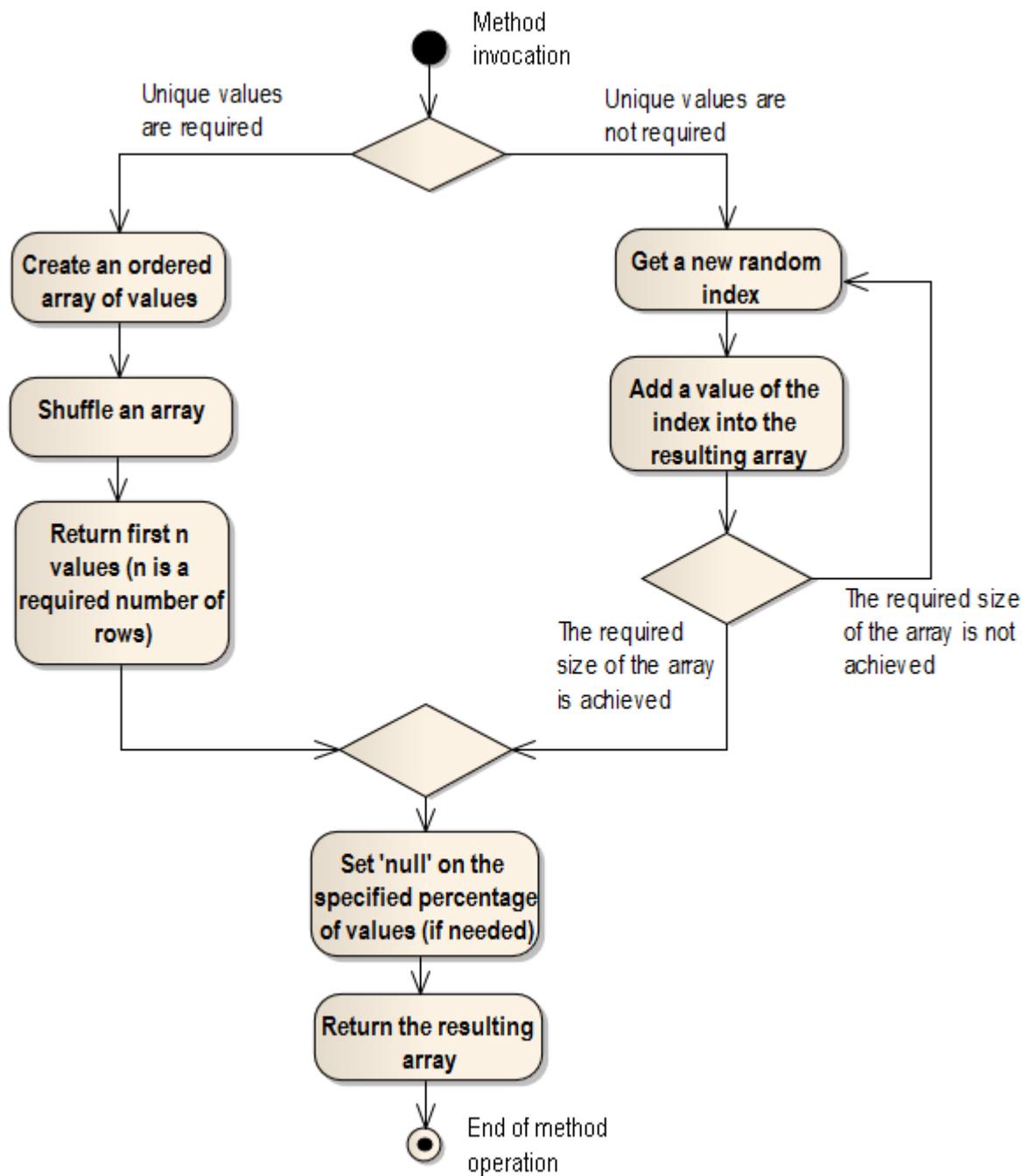


Fig.16. The algorithm of an integer data generator operation

The algorithm of a data type «Date and time» data generator NextSet method via UML activity diagram is shown on fig.17.

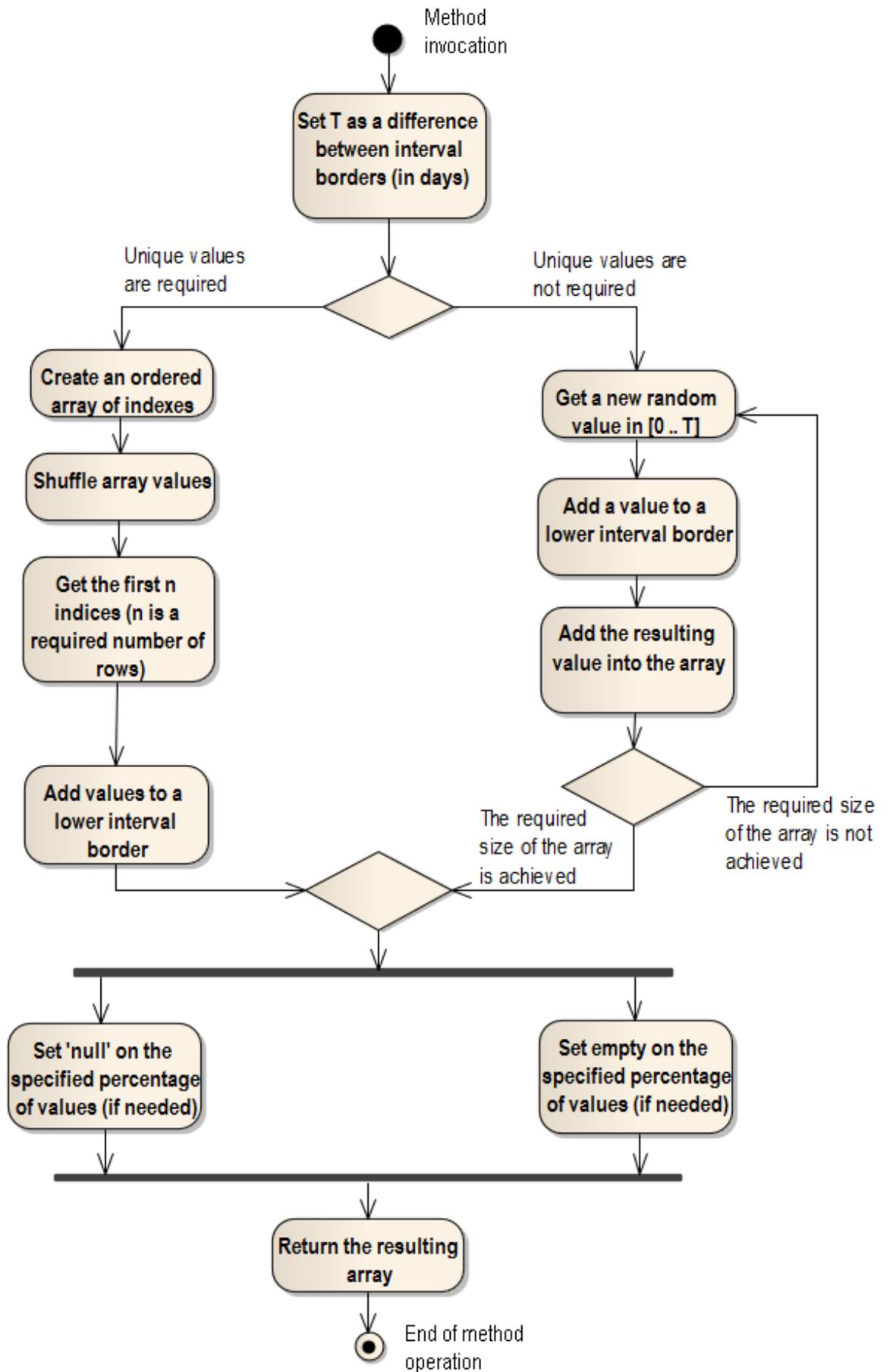


Fig.17. The algorithm of date and time generator operation

The algorithm of a random text generator operation via UML activity diagram is shown on fig. 18.

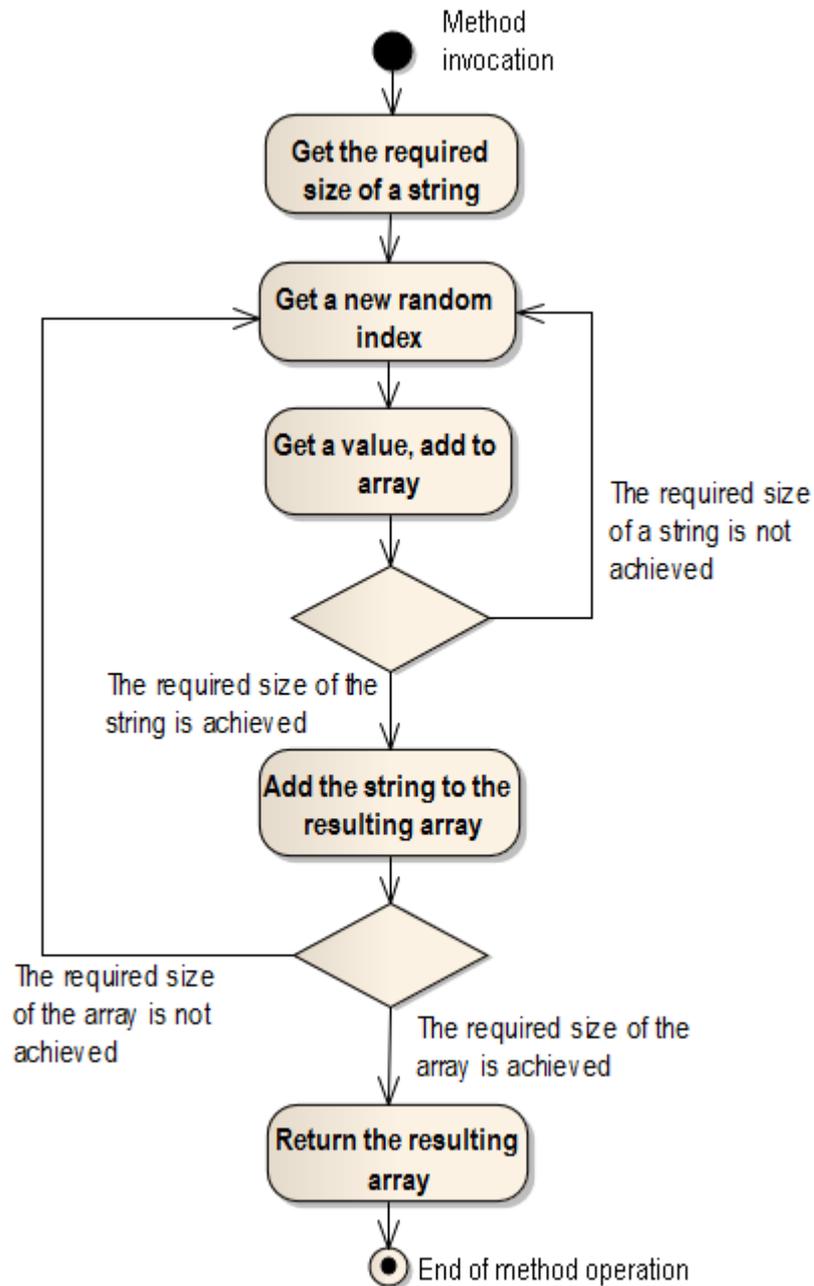


Fig. 18. The algorithm of a random text generator operation

2.5 The service database

The main purpose of using service built-in database (DB) is a storage of user settings and program parameters.

Relational database management system (DBMS) SQLite was chosen to manage the service database in the program system.

SQLite is a compact embedded relational database. The library's source code is transferred to the public domain. The word «embedded» means that SQLite does not use the client-server paradigm. It means that the SQLite engine is not a separately running process that the program interacts with but it provides the library with which the program is linked. The SQLite engine becomes an integral part of the program. Thus, the exchange protocol is used to call the functions (API) of the SQLite library [1].

The service database has a simple structure, which is shown on fig.19.

Table Name	Columns	Primary Key
FKList	FKName TEXT, TableName TEXT, ColumnName TEXT, RefTableName TEXT, RefColumnName TEXT	None
TypeAutoAssignRules	Priority INT, GenType TEXT, Rule TEXT	None
ColumnAttributes	ID INT, TABLE_NAME TEXT, COLUMN_NAME TEXT, DATA_TYPE TEXT, IS_NULLABLE TEXT, CHARACTER_MAXIMUM_LENGTH INT	ID
CustomGeneratorsList	CGName TEXT, Path TEXT	None

Fig.19. Tables of the service DB

The database tables has the following assignment:

- FKList – is a storage of a list of foreign keys.
- TypeAutoAssignRules – is a storage of rules for auto-assignment of generators.
- ColumnAttributes – is a storage of properties of table columns.
- CustomGeberatorsList – is a storage of the list of user data generators and paths to files with the values of these generators.

2.7 Compliance with the integrity constraints of the target database

Integrity of the generated data in terms of the relational data model entities in the developed system is reduced to the requirement of a uniqueness of the primary key attribute.

Generating unique values mechanism by exhibiting the properties of data generators is implemented in the developed system. Thus, the problem is reduced to prevent the generation of duplicate values by software assigning a specific value to a corresponding property of the data generator object.

Diagram of the system behavior when functioning with this restriction is shown on fig. 20.

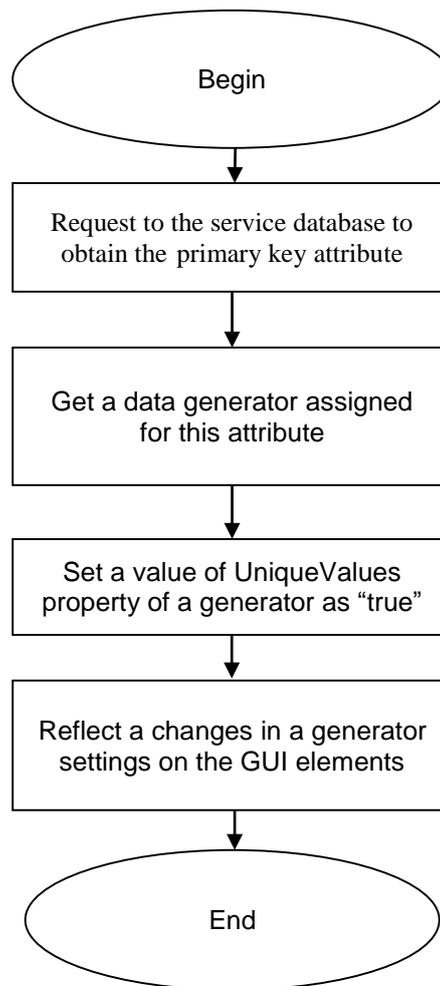


Fig. 20. The algorithm of functioning with integrity constraints of database entities

For the references, the requirement of integrity is reduces to the problem of a limiting the possible data set of the attribute of the child relationship. There must be a cortege with the same value of the key attribute in the parent relation.

Logic must be implemented in the software system as it allows the system to achieve its goal, without violating this integrity constraint.

Diagram of the system behavior when functioning with this restriction is shown on fig. 21.

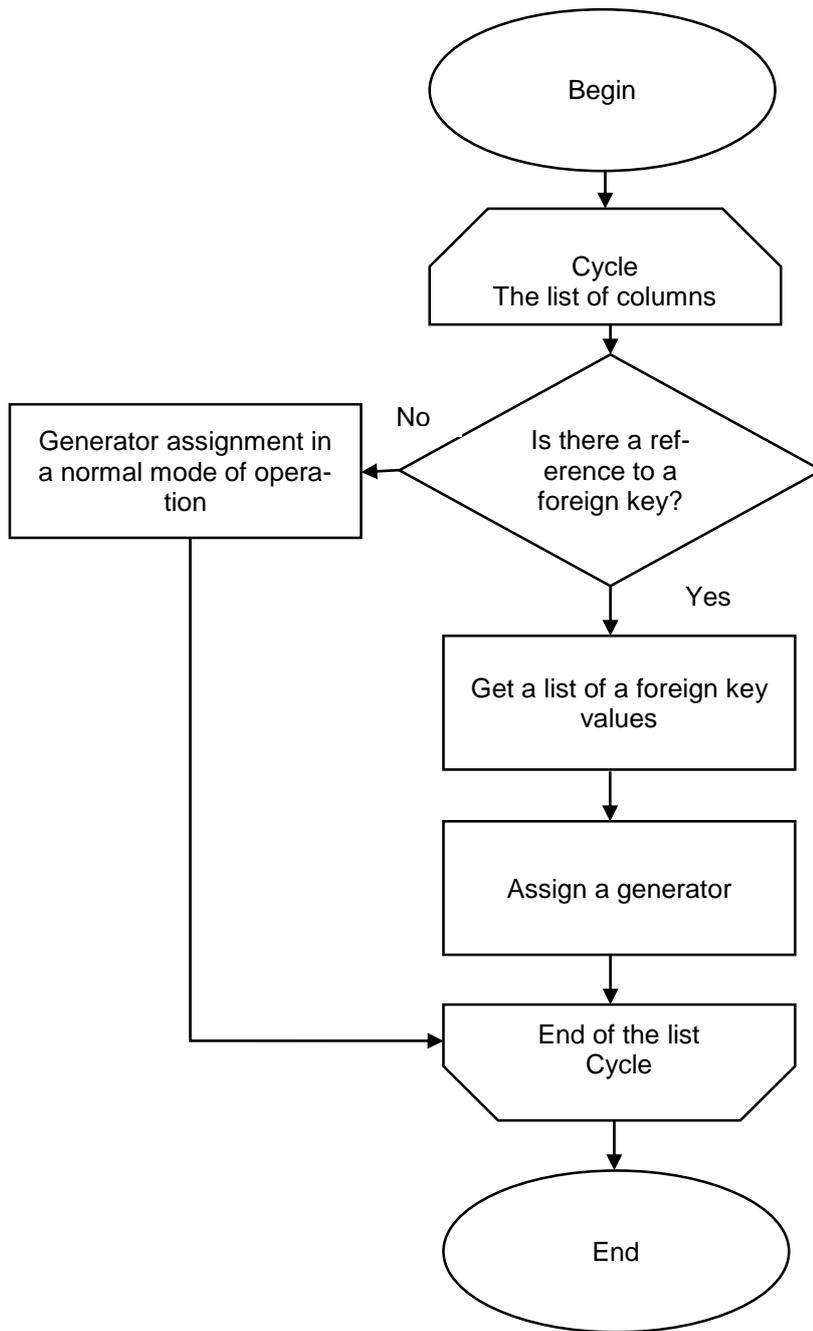


Fig. 21. The algorithm of the program operation when assigning a generator using a foreign key

References

1. SQLite – Wikipedia. – Available at: URL: <https://ru.wikipedia.org/wiki/SQLite/> (accessed 11 apr 2017)