

КУЗНЕЦОВ И. А., ЛАПИНА Н. Н.
ИНФОРМАЦИОННАЯ СИСТЕМА УЧЕТА ОФОРМЛЕНИЯ И
ВЫПОЛНЕНИЯ ЗАКАЗОВ ИСПЫТАТЕЛЬНОЙ ЛАБОРАТОРИИ
СТРОИТЕЛЬНЫХ МАТЕРИАЛОВ И ГРУНТОВ

УДК 004.62:002.5, ВАК 05.13.01, ГРНТИ 50.53.17

Информационная система учета
оформления и выполнения заказов
испытательной лаборатории
строительных материалов и грунтов

Accounting information system design
in order for a test laboratory of
construction materials and soil

И. А. Кузнецов¹, Н. Н. Лапина²

I. A. Kuznetsov¹, N. N. Lapina²

¹Самарский государственный
аэрокосмический университет
им. С. П. Королёва, г. Самара

¹Samara state aerospace University.
S. P. Korolev, Samara

²Филиал «Нижегородское управление
по организации общественного
питания» ООО «Газпром питание»,
г. Нижний Новгород

²The branch "Nizhny Novgorod
management on the organization of
public catering" LLC "Gazprom food",
Nizhny Novgorod

*В статье описан процесс учета в
испытательной лаборатории и
разработка информационной системы
для его автоматизации, проведен
поиск и обзор аналогов. Представлена
логическая модель уровня управления
данными, диаграмма прецедентов для
системы, описание созданного
прототипа.*

*The article describes the process of
accounting in the testing laboratory and
the development of an information sys-
tem for its automation, a search and re-
view of analogues is made. Logical
model of the data management layer are
presented, a use case diagram for the
system, a description of the prototype
created.*

Ключевые слова: система учета,
учет заказов, автоматизация бизнес-
процесса, испытательная
лаборатория, строительные
материалы, грунты.

Keywords: accounting system, or-
der accounting, business process auto-
mation, testing laboratory, building ma-
terials, soils.

Введение

Испытательная лаборатория строительных материалов и грунтов (ИЛ-СМиГ) занимается проведением испытаний строительных материалов и грунтов. Изначально все процессы организации, начиная от оформления заявки на испытания и до оформления протокола, а также подсчета цены заказа и предоставления списков на составление актов, в данной лаборатории никаким образом не

автоматизированы. Здесь имеет место и большое количества ручного труда, и бумажной документации, которую неудобно хранить и упорядочивать – для того, чтобы полностью оформить один заказ, нужно занести данные во множество журналов, при этом продублировав некоторые записи.

Кроме того, в настоящее время в лаборатории в электронном виде хранятся дубликаты протоколов с приложениями, создаваемые с помощью офисных программ (Word, Excel), используемые для печати. Но из-за организации процесса, создание электронных экземпляров представляет собой лишнюю работу по переносу данных из имеющихся журналов. Подобный способ организации работы не позволяет гибко организовывать таблицы результатов для заказчиков, затрудняется поиск нужной информации. Также время- и трудозатраты выше, чем при наличии полноценной ИС для управления данным процессом.

Целью данной работы является разработка ИС для ведения учета оформления и выполнения заказов в указанной испытательной лаборатории, что позволит избавиться от многих недостатков неавтоматизированных процессов.

Обзор аналогов

В смежной области инженерно-геологических изысканий и испытаний строительных материалов программного обеспечения на данный момент не существует, либо оно используется в закрытых кругах и не распространено. В сфере инженерно-геологических изысканий присутствуют следующие представители коммерческого ПО для нужд инженерной геологии [1]:

- CREDO ГЕОЛОГИЯ (ООО «Кредо-Диалог», credo-dialogue.ru);
- GEOSimple (Дёмин Сергей Андреевич, geosimple.ru);
- EngGeo (Мелихова Татьяна Юрьевна, enggeo.net);
- GEOTECH Геолог + GeoDraw (ООО «Юнис-Юг», www.uniservice-europe.co.uk);
- GeoniCS GEODirect (АО «СиСофт Девелопмент», www.csdev.ru).

Это комплексные программные системы, которые, однако, хоть и имеют возможность обработки данных лабораторных испытаний и позволяют экспортировать данные, например, в MS Excel и MS Word, специализируются на более комплексной деятельности и не подходят даже для частичной автоматизации процессов ИЛСМиГ. Так, все они обладают широким функционалом и, безусловно, являются мощными инструментами для инженерно-геологических изысканий, однако не подходят для формирования заявок и прочей деятельности лаборатории.

Также существует программа Лаборатория, позволяющая упростить расчеты при проведении измерений для грунтовых проб и избавиться от части журналов с результатами, однако она не позволяет оформлять протоколы испытаний и прочую документацию и сфокусирована именно на облегчении проведения работ с грунтами [2]. На рисунке 1 представлена организация пользовательского интерфейса программы Лаборатория (версия 3.0). Он состоит из трех основных областей:

- 1) область скважин;

- 2) область проб;
- 3) область исходных данных.

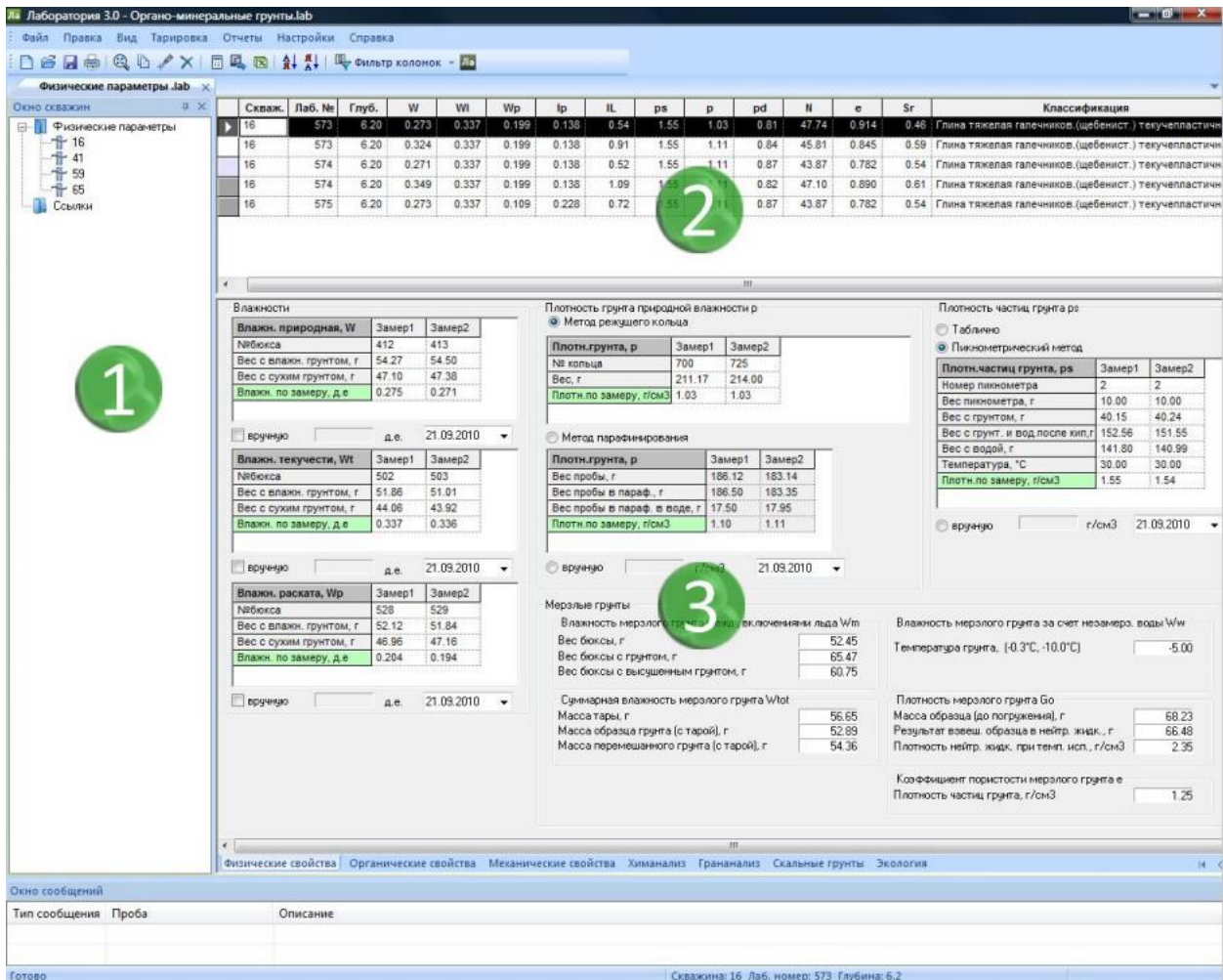


Рисунок 1. Организация пользовательского интерфейса ПО «Лаборатория»

В области испытания строительных материалов относительно похожего на разрабатываемое ПО еще меньше: подавляющее количество программных продуктов предназначено для взаимодействия с испытательной техникой, такой как, например, гидравлические прессы, что не может считаться близким аналогом для разрабатываемой в рамках данной работы информационной системы. Среди подобного программного обеспечения можно выделить Bluehill, позволяющее, кроме всего прочего, гибко настраивать вывод результатов [3].

Исходя из вышеизложенного, на данный момент для планируемой системы нет близких аналогов или же они не распространены широко. Имеющееся же программное обеспечение сфокусировано на отличных от создаваемой ИС задачах, а также не покрывает достаточно сферу деятельности лаборатории. Так что можно сказать, что данная система разрабатывается для не занятой на данный момент ниши.

Постановка задачи

Исходя из организации неавтоматизированных процессов и отсутствия подходящих аналогов, становится видна потребность написания такой информационной системы, которая будет покрывать деятельность лаборатории, позволяя тем самым автоматизировать процесс учета оформления и выполнения заказов испытательной лаборатории, работающей со строительными материалами и грунтами. Система должна позволять создавать, регистрировать и хранить заявки на испытания, формировать протоколы и списки на акты, а также агрегировать результаты испытаний и брать часть расчетов на себя. Все это должно позволить уменьшить количество бумажной работы, облегчить процессы учета и сделать менее критичным человеческий фактор.

Выбор средств разработки

В силу того, что проводится индивидуальная разработка с минимальной коммерциализацией, было решено использовать среду Microsoft Visual Studio в версии Community Edition. Языком программирования выбран C#. Выбор среды и языка также обусловлен тем, что C# – гибкий объектно-ориентированный язык, поддерживающий многие модели разработки, сборку мусора посредством CLR, а также очень удобный в разработке. Кроме того, в Visual Studio удобно реализовано создание многозвенных приложений, что также повлияло на выбор.

Для реализации базы данных же была выбрана PCУБД Microsoft SQL Server в силу близкой интеграции с Visual Studio, а также из-за поддержки процедурного расширения языка SQL T-SQL.

В качестве средства взаимодействия с БД было решено использовать LINQ to SQL, поддерживаемый Visual Studio за счет хорошей масштабируемости, возможности работы с реляционными данными как с объектами, отсутствия необходимости явно встраивать в код SQL-запросы и возможности с его помощью реализации паттерна MVC.

Для построения диаграмм и прототипирования интерфейса использовалась программа Visual Paradigm. Выбор обусловлен поддержкой UML нотации, возможностью создавать самые разнообразные диаграммы и создавать прототипы интерфейса.

Реализация

Перед непосредственным началом разработки было составлено техническое задание. Оно стало основным документом, направляющим весь процесс разработки ИС. Кроме прочего, ТЗ содержит в себе характеристику объектов автоматизации и комплекс требований к системе, в том числе функциональных. После составления ТЗ было принято решение к началу непосредственной разработки информационной системы.

Также наряду с ТЗ была создана модель бизнес-процесса, составлены контекстные диаграммы «как есть» и «как будет», а также проведена декомпозиция и построена диаграмма потоков данных для данного процесса (DFD). Контекстную диаграмму процесса (так как взаимодействие с внешними сущностями не

меняется, контекстные диаграммы «как есть» и «как будет» одинаковы) и DFD можно увидеть ниже на рисунках 2 и 3 соответственно.

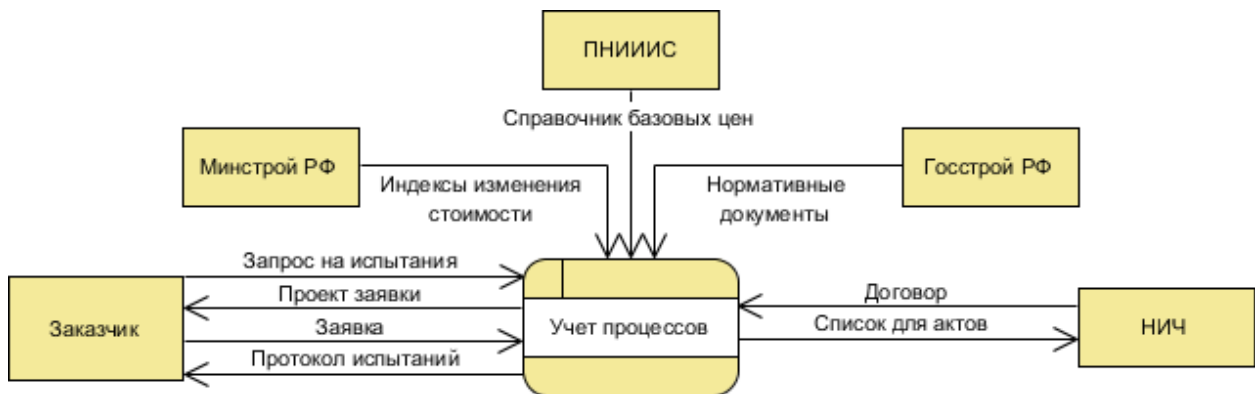


Рисунок 2. Контекстная модель процесса

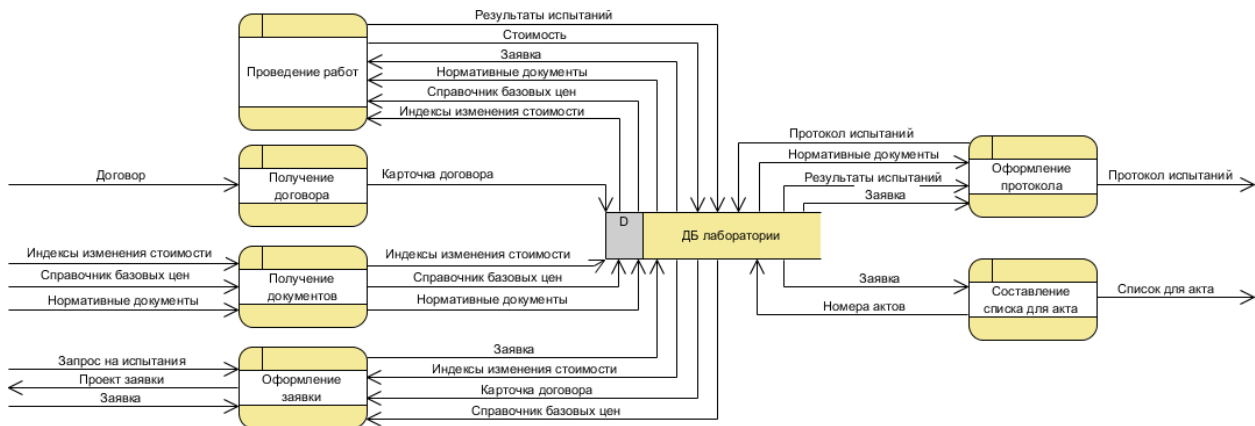


Рисунок 3. DFD 1-го уровня БП «Учет оформления и выполнения заказов испытательной лаборатории» «Как будет»

Построение логической модели базы данных – один из первых шагов при ее проектировании, включающий в себя представление сущностей базы и уточнение их атрибутов, установление связей между сущностями. Все это строится без привязки к какой-либо конкретной реализации, в потому в логической модели отсутствует сопоставление атрибутов сущностей типам данных. После выявления сущностей, определения их связей, объединения сущностей со связями один-к-одному и создания слабых при связях многие-ко многим, в конечной модели получилось 26 сущностей. Данная модель представлена на рисунке 3 в виде диаграммы сущность-связь.

Далее на основе логической модели была построена физическая модель базы данных, рассматривающая БД в контексте средств реализации базы. Фактически, от логической она отличается только прописанными ограничениями СУБД (например, типы данных, разрешение значений NULL). После чего была реализована собственно БД.

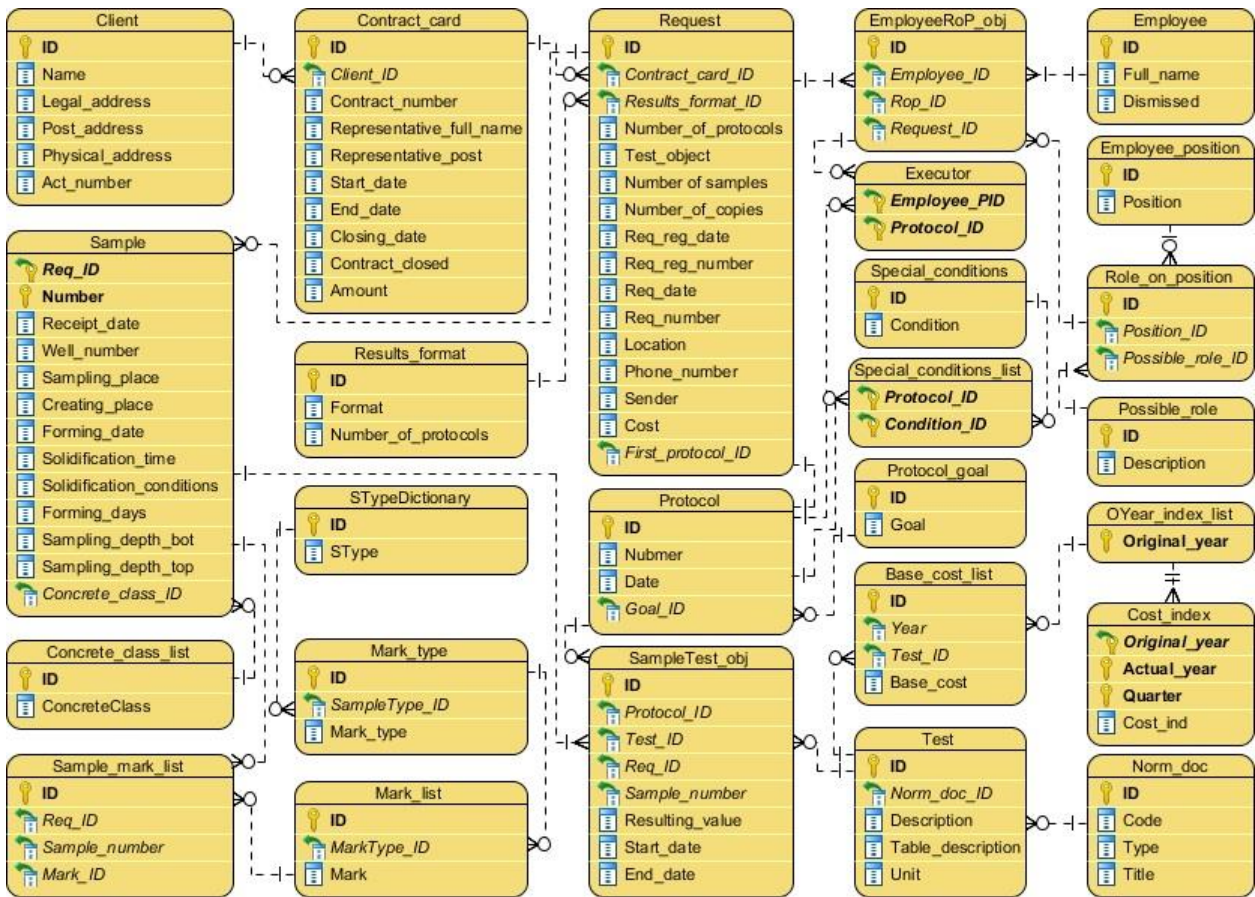


Рисунок 3. Логическая модель БД (ERD)

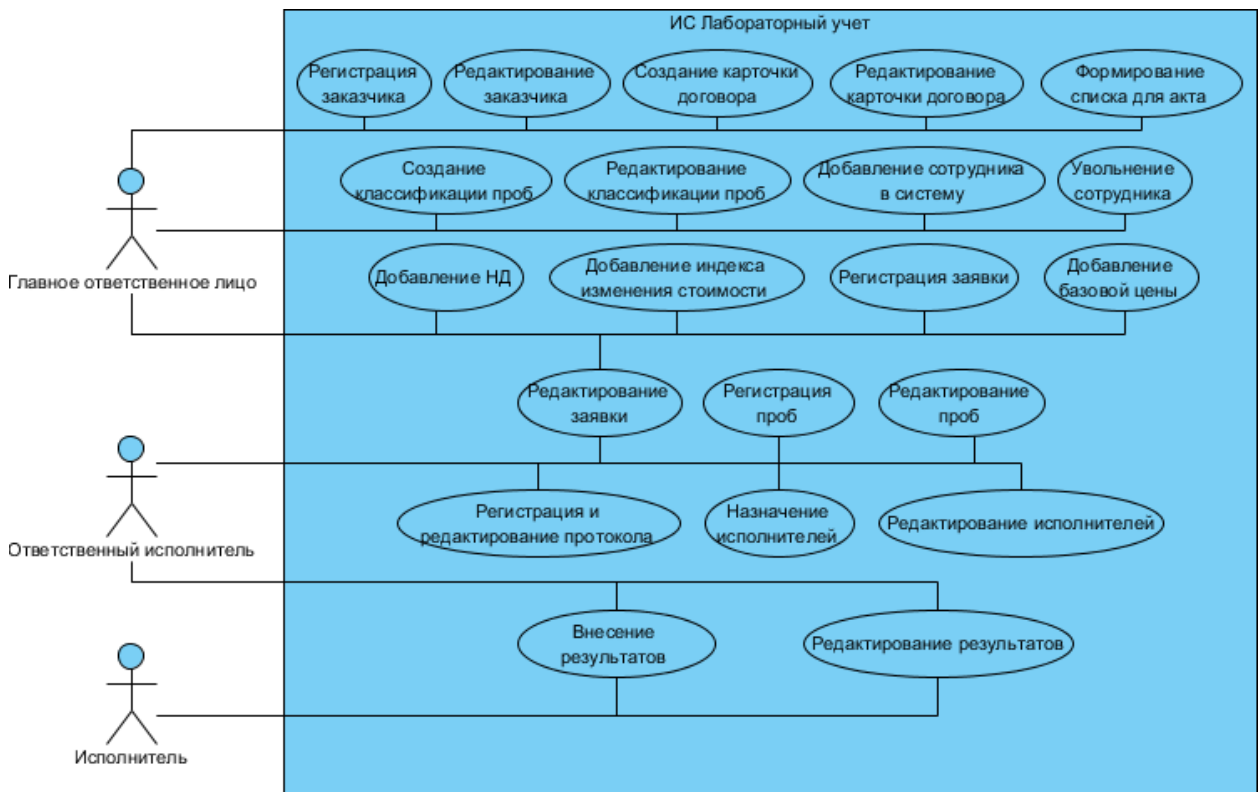


Рисунок 4. Диаграмма прецедентов

Кроме создания базы данных на основе построенных моделей, было создано представление поведения системы в виде диаграммы прецедентов (рис. 4), на которой указаны типовые прецеденты работы системы в соответствии с пользовательскими ролями.

После определения прецедентов работы системы, каждый из них был подробно описан, и для всех прецедентов созданы прототипы интерфейса.

После завершения работ с интерфейсами, наступила очередь реализации функциональных требований, установленных ТЗ.

Так как инструментом взаимодействия с БД был выбран LINQ, то был использован один класс-контроллер, занимающийся отображением БД ее работой с ней. Он был вынесен в отдельный проект в рамках решения, что позволит в процессе внедрения вынести его на отдельный сервер.

На данный момент основной интерфейс выглядит следующим образом: на форме приложения располагается элемент MenuStrip, навигация по функциям приложения осуществляется путем перемещения по пунктам меню. На главной панели приложения расположены элементы управления заказчиками – кнопки добавления и удаления заказчиков и карт договоров, а также таблицы (DataGridView), показывающие и позволяющие редактировать информацию по ним. Все это можно увидеть на рисунке 5.

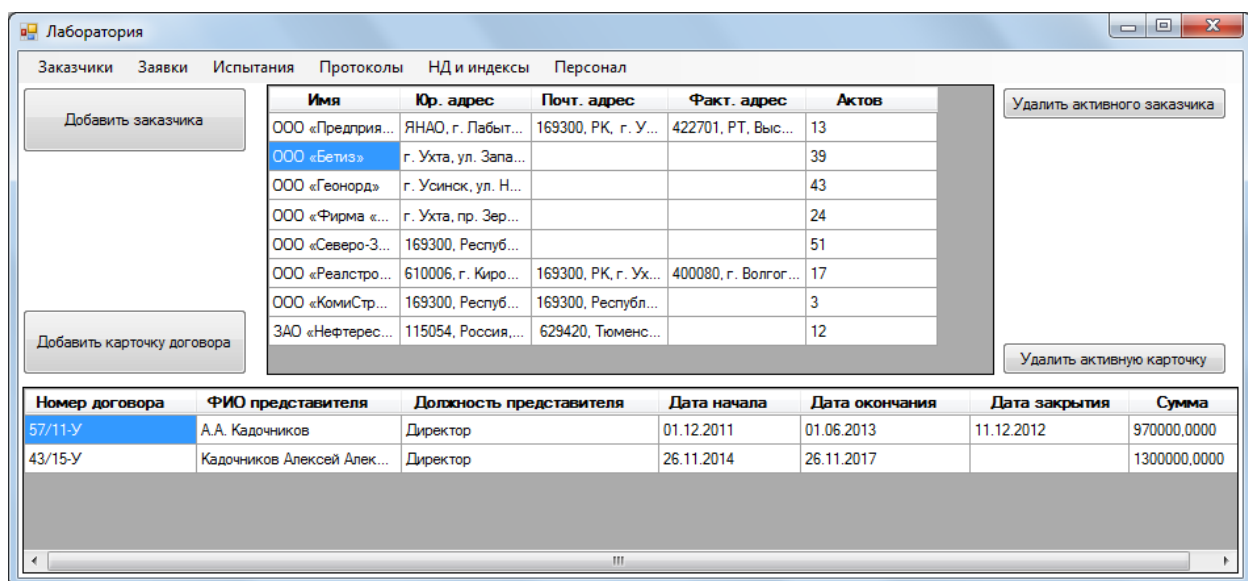


Рисунок 5. Начальная панель приложения

Остальные панели содержат соответствующие данные, представленные схожим с основной панелью способом (рис. 6).

При запуске приложения все данные о заказчиках, внесенных в систему загружаются в таблицу и отображаются на главной панели. Для того, чтобы добавить нового заказчика в систему, необходимо нажать на кнопку «Добавить заказчика». В ответ система отображает модальное окно (рис. 7), после чего в случае подтверждения добавления заказчика, обновляет данные в БД и актуализирует представление.

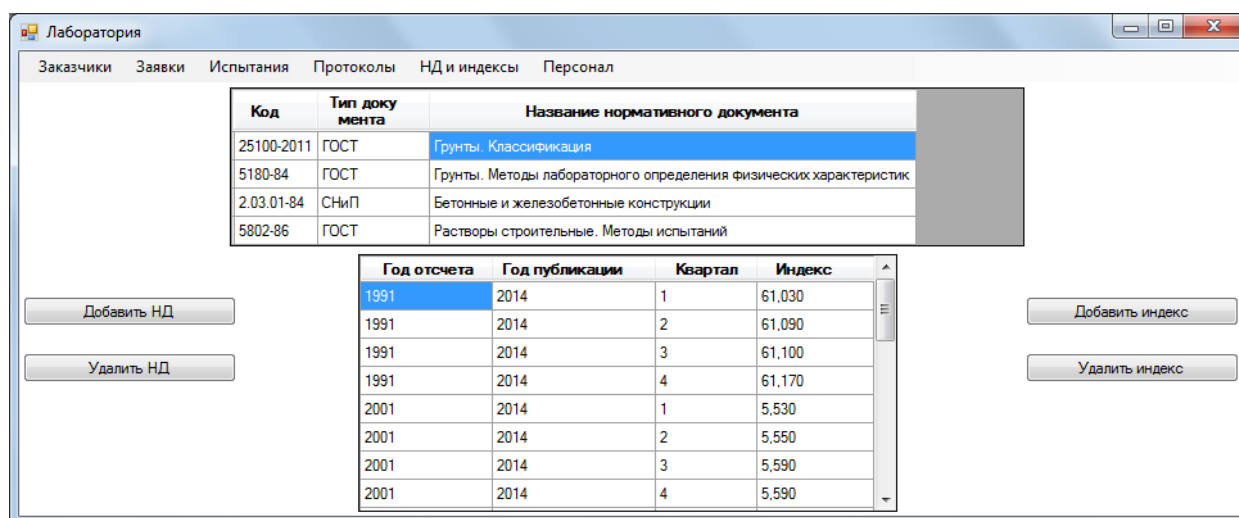


Рисунок 6. Панель управления нормативными документами и индексами

Имя заказчика* 123123

Юр. адрес* 456456

Почтовый адрес 789789

Фактический адрес 147147

поля, помеченные *, обязательны к заполнению

Отмена Добавить заказчика

Рисунок 7. Окно добавления заказчика

После добавления записи в случае необходимости можно редактировать данные заказчика прямо из основного окна.

Заключение

В процессе работы была описана предметная область, изучены аналоги, сформулированы требования к продуктам подобного рода. Выполнено создание прототипа, реализующего значительную часть запланированного функционала. После успешного завершения дальнейшей разработки и превращения прототипа в конечный продукт можно будет внедрить его на предприятие, выполнив тем самым поставленную изначально задачу автоматизации бизнес-процесса.

Список литературы

1. Прохачев М. Обзор ПО для инженерной геологии // *GeoInfo* – информационный ресурс для инженеров-изыскателей [Электронный ресурс]. Режим доступа: http://www.geoinfo.ru/includes/periodics/programmnoe_obespechenie/2017/0329/000012805/detail.shtml.
2. Лаборатория // ПСС Грайтек [Электронный ресурс]. Режим доступа: <http://www.pss.spb.ru/products/Yuniservis/Laboratory.html>.
3. Instron Bluehill Universal Software // Instron [Электронный ресурс]. Режим доступа: <http://www.instron.ru/ru-ru/products/materials-testing-software/bluehill-universal>.
4. Microsoft, Руководство Microsoft по проектированию архитектуры приложений, 2-е издание. 2009. 529 с. Режим доступа: download.microsoft.com/documents/rus/msdn/ры_приложений_полная_книга.pdf.
5. Фримен Э., Сьерра К., Бейтс Б. Паттерны проектирования. СПб. : Питер, 2012. 656 с.

List of references

1. Prokhachev M. “Overview Software for Engineering Geology”, *GeoInfo – Information resource for prospecting engineers*, http://www.geoinfo.ru/includes/periodics/programmnoe_obespechenie/2017/0329/000012805/detail.shtml.
2. “Laboratory”, *PSS Graitec*, <http://www.pss.spb.ru/products/Yuniservis/Laboratory.html>.
3. “Instron Bluehill Universal Software”, *Instron*, <http://www.instron.ru/en-us/products/materials-testing-software/bluehill-universal>.
4. Microsoft, Microsoft Application Architecture Guide, 2nd Edition, MSDN, <https://msdn.microsoft.com/en-us/library/dd673617.aspx>.
5. Freeman E., Sierra K., Bates B. *Head First Design Patterns*, O'Reilly Media, Inc, Sebastopol, California, United States, 2004, 676 p.