

Березинец Я.В.

*Научный руководитель: к.т.н., доцент каф. САПР А.А. Быков  
Муромский институт (филиал) федерального государственного образовательного  
учреждения высшего образования «Владимирский государственный университет  
имени Александра Григорьевича и Николая Григорьевича Столетовых»  
602264, г. Муром, Владимирская обл., ул. Орловская, 23  
E-mail: berezinec.yaroslav@mail.ru*

### **Проектирование автоматизированной информационной системы (АИС) ремонта и технического обслуживания металлорежущего оборудования в механо-сборочном цехе**

Общей концепцией поддержания оборудования в исправном состоянии и постоянной работоспособности является применение системы планово-предупредительного ремонта (ППР), которая законодательно закреплена в ГОСТ для внедрения на всех предприятиях страны [1]. Составлением графика ППР занимается отдел главного механика (ОГМ). Система реализуется для механо-сборочного цеха и ОГМ АО «МЗ РИП», с учетом используемых на предприятии технологий и нормативных документов.

Составление графика ППР основано на учете и анализе большого объема исходной информации как известной априорно из эксплуатационной документации, так и поступающей в процессе функционирования оборудования. Например, дата запуска оборудования в эксплуатацию и рекомендованный состав и технология его технического обслуживания, конструктивные особенности конкретного подтипа оборудования в соответствии с их назначением и режимами эксплуатации, а также состав и продолжительность аварийных ремонтов, проведенные вне графика ППР. Важными факторами является присутствие подготовленных специалистов необходимой квалификации и их текущая загруженность, а также наличие запасных частей и расходных материалов. Большой объем исходной и текущей информации, значительное время на ее сбор и обработку не позволяют вручную составить безошибочный график ППР и реализовать его с высоким качеством и минимальными финансовыми затратами.

Для решения рассмотренной проблемы была поставлена задача разработать автоматизированную информационную систему (АИС) [2], которая позволит составлять графики ППР, а также собирать всю необходимую информацию по оборудованию механо-сборочного цеха, плановым и аварийным ремонтам, запасным частям и расходным материалам, применяемым в ходе ремонтов. В системе будет реализовано разделение функционала для двух ролей – пользователь (инженер и механик) и администратор. Инженер сможет изменять данные Администратор сможет вносить изменения в справочники (таблицы, недоступные для изменения пользователям, некоторые данные из которых пользователи могут только просматривать) и список пользователей и работать с резервными копии базы данных (БД). При изменении данных и работе с резервными копиями основные действия сохраняются в лог-файлы.

Разрабатываемая АИС будет обладать следующими функциями:

- добавление, удаление, изменение моделей оборудования;
- добавление, удаление, изменение экземпляров оборудования;
- добавление, удаление, изменение масел;
- добавление, удаление, изменение запчастей;
- добавление, удаление, изменение СОЖ;
- учёт использования масел, СОЖ и запчастей при ремонте оборудования;
- построение отчетов по графикам ППР и ремонтам оборудования согласно формам ОГМ001, ОГМ002, ОГМ005, ОГМ008 СТО МВИА.460009.020 и опираясь на СТО МВИА.460009.024-2016 АО «МЗ РИП» [3,4];
- экспорт отчетов в файлы xlsx;
- поиск по базе данных и фильтрация при поиске;
- резервное копирование данных из БД;
- настройка пользовательского интерфейса

- планирование графиков ППР;
- внесение правок в график ППР: возможность добавления аварийных ремонтов и регенерация графика ППР.

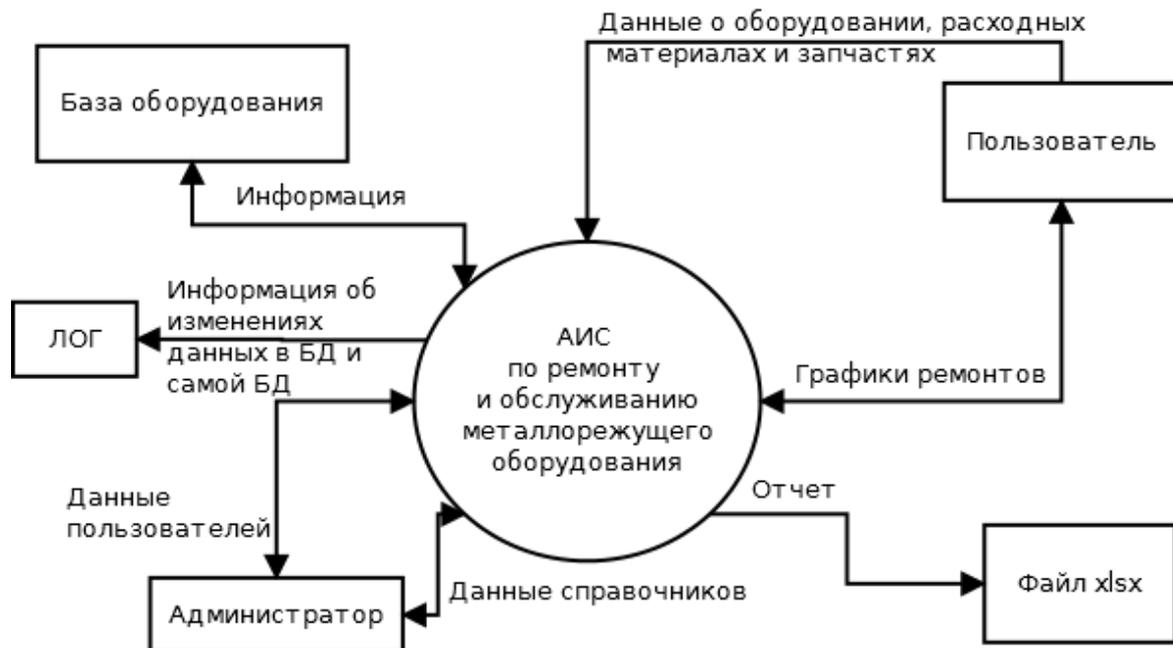


Рис.1 – Контекстная диаграмма

Для разработки будет использован язык программирования С#. Для работы с базами данных будет использован SQL Server в комплексе с SQL Server Management Studio (SSMS). Оба программных продукта используются на АО «МЗ РИП» и являются интегрированными в производство, что позволит существенно упростить процесс внедрения разрабатываемой АИС. Возможна реализация авторизации с помощью Active Directory.

На данный момент разрабатываемая АИС представляет собой многооконное десктопное приложение (структура на рис.1), однако в дальнейшем планируется ее преобразование в систему на основе трехуровневой (трехзвенной) архитектуры.

### Литература

1. Ящура А.И. Система технического обслуживания и ремонта общепромышленного оборудования: Справочник. – М.:Изд-во НИЦ ЭНАС, 2006. – 360 с.
2. Бурков А.В. Проектирование информационных систем в Microsoft SQL Server 2008 и Visual Studio 2008 [Электронный ресурс]/ Бурков А.В.— Электрон. текстовые данные.— М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016.— 310 с.
3. ГОСТ Р ИСО 9000-2015 Системы менеджмента качества. Основные положения и словарь - М.: Стандартинформ, 2015. – 16 с.
4. ГОСТ РВ 0015-002-2012 Система разработки и постановки продукции на производство военной техники. Системы менеджмента качества. Общие требования. – М.: Стандартинформ, 2012. – 38 с.

Васильцов И.И.

*Научный руководитель: к.т.н., доцент А.А. Быков  
Муромский институт (филиал) федерального государственного образовательного  
учреждения высшего образования «Владимирский государственный университет  
имени Александра Григорьевича и Николая Григорьевича Столетовых»  
602264, г. Муром, Владимирская обл., ул. Орловская, 23  
E-mail: ilya.vasiltsov@mail.ru*

### Использование технологии виртуализации для защиты кода программной платформы .NET Framework

.NET Framework — программная платформа, выпущенная компанией Microsoft в 2002 году. Основой платформы является общезыковая среда исполнения Common Language Runtime (CLR), которая подходит для разных языков программирования. Функциональные возможности CLR доступны в любых языках программирования, использующих эту среду.[1]

Использование программной платформы позволяет разработчику облегчить создание и поддержку технически сложных или нагруженных проектов. Библиотека Framework Class Library (FCL), содержащая несколько тысяч объектных классов .NET, предоставляет основные программные модули, на основе которых разработчиками реализуются специфические для проекта компоненты.

Основной идеей при разработке .NET Framework являлось обеспечение свободы разработчика за счёт предоставления ему возможности создавать приложения различных типов, способные выполняться на различных типах устройств и в различных средах.

После выполнения компиляции код .net приложений(сборок) содержит набор метаданных, описывающих сборку, ее типы и члены, а также IL-код (сокр. От CIL - Common Intermediate Language), который реализует все члены типов. Для выполнения программы на машине конечного пользователя должна быть установлена среда CLR (Common Language Runtime (CLR)). Встроенный в CLR JIT-компилятор «на лету» (just in time) преобразует промежуточный байт-код в машинные коды нужного процессора[1]. (Рис.1)

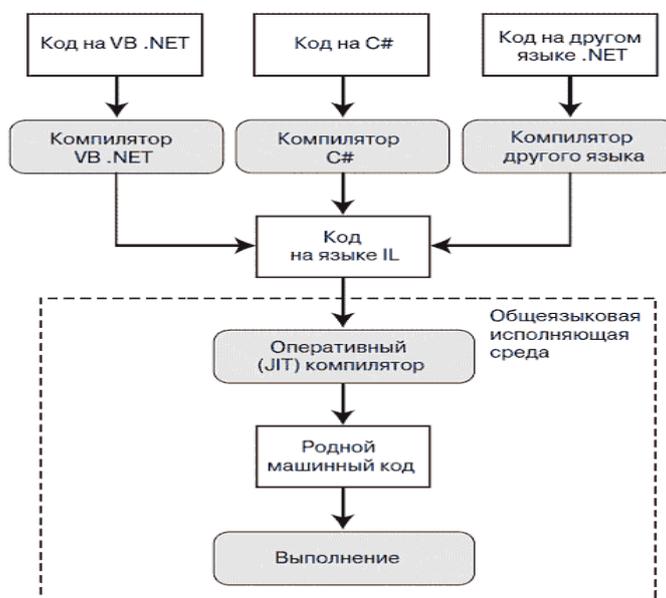


Рис.1. Компиляция и выполнение программ на платформе .Net framework

Скомпилированный код для .NET Framework помещается в логические единицы, называемые сборками (assembly). Компоненты сборки могут находиться в одном или нескольких файлах.

Каждая сборка содержит:

- Манифест
- Метаданные типов
- IL – код
- Ресурсы

Программы, разрабатываемые с помощью .NET Framework, создаются для среды CLR.

Над сборками можно провести операцию декомпиляции – получения исходного кода.

Причины возможности декомпиляции:

- Большое количество информации о сборке в метаданных
- IL – код является абстракцией более высокого уровня, чем машинный код
- Компиляторы языков .net framework выполняют меньше оптимизации кода, чем машинный код компиляторов, преобразующих программу в машинный код.

К настоящему времени разработано множество различных декомпиляторов, позволяющих просматривать исходный код любых .net сборок в виде эквивалентного языка c# или в виде набора IL инструкций. Возникает фундаментальная проблема, а именно: написанная человеком программа может быть человеком же и понята, проанализирована, разобрана. Распространяя свое программное обеспечение, разработчики неизбежно раскрывают свои наработки и алгоритмы, которыми может воспользоваться любой желающий.

Для решения возникшей проблемы необходим алгоритм, который может до неузнаваемости изменить код программы, при этом сохраняя ее функциональность.

В ходе выполнения исследовательской работы было разработано приложение, позволяющее предотвратить возможность декомпиляции.

Виртуализация — предоставление набора вычислительных ресурсов или их логического объединения, абстрагированное от аппаратной реализации, и обеспечивающее при этом логическую изоляцию друг от друга вычислительных процессов, выполняемых на одном физическом ресурсе. [2]

Одно из применений виртуализации заключается в симуляции процессора. Это, так называемая, P-code(или pseudo-code) машина. P-code–это машинный язык, который выполняется на виртуальной машине, а не на реальном оборудовании.

Разработанное программное обеспечение использует технику виртуализации кода, позволяя таким образом значительно усложнить анализ защищённых программ. Приложение выполняет трансляцию кода исходной функции, представляющей собой код на языке IL в сформированный случайным образом байт-код виртуальной машины приложения. Исходный код функции удаляется и заменяется вызовом виртуальной машины.

### Литература

1. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C#. 4-е изд. — СПб.: Питер, 2013. — 896 с.
2. URL: <https://ru.wikipedia.org/wiki/Виртуализация> [Электронный ресурс] Дата посещения 23.04.18
3. URL: [https://ru.wikipedia.org/wiki/.NET\\_Framework](https://ru.wikipedia.org/wiki/.NET_Framework)[Электронный ресурс] Дата посещения 23.04.17

Васяева Д.О.

*Научный руководитель: к.т.н. Белякова А.С.*

*Муромский институт (филиал) федерального государственного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»  
602264, г. Муром, Владимирская обл., ул. Орловская, 23  
E-mail: v-dasha95@yandex.ru*

## **Исследование использования технологии OLAP в системе поддержки принятия решений**

Информационные системы масштаба организации, как правило, содержат приложения, предназначенные для комплексного многомерного анализа данных. Такой анализ в конечном итоге призван содействовать принятию решений. Нередко эти системы так и называются – системы поддержки принятия решений. [1]

Системы поддержки принятия решений обычно обладают средствами предоставления пользователю агрегатных данных для различных выборок из исходного набора в удобном для восприятия и анализа виде. [1]

Как правило, такие агрегатные функции образуют многомерный (и, следовательно, нереляционный) набор данных (нередко называемый гиперкубом или метакубом), оси которого содержат параметры, а ячейки – зависящие от них агрегатные данные – причем храниться такие данные могут в реляционных таблицах. [3]

Вдоль каждой оси данные могут быть организованы в виде иерархии, представляющей различные уровни их детализации. Благодаря такой модели данных пользователи могут формулировать сложные запросы, генерировать отчеты, получать подмножества данных.

Технология комплексного многомерного анализа данных получила название OLAP (On-Line Analytical Processing).

Такая технология хорошо подходит для использования её в системе оценки состояния здоровья и подбора лекарственных средств.

Существуют требования к приложениям для многомерного анализа, основанные на требованиях, изложенных Коддом в 1995 году:

- предоставление пользователю результатов анализа за приемлемое время (обычно не более 5 с), пусть даже ценой менее детального анализа;
- возможность осуществления любого логического и статистического анализа, характерного для данного приложения, и его сохранения в доступном для конечного пользователя виде;
- многопользовательский доступ к данным с поддержкой соответствующих механизмов блокировок и средств авторизованного доступа;
- многомерное концептуальное представление данных, включая полную поддержку для иерархий и множественных иерархий (это – ключевое требование OLAP);
- возможность обращаться к любой нужной информации независимо от ее объема и места хранения. [4]

Язык запросов, использующийся в технологии OLAP – MDX отдаленно напоминает язык запросов SQL, но и SQL так же используются, например, в системе OLAP анализа Oracle OLAP Option. [2]

Существует ряд нереляционных СУБД для работы с OLAP технологией:

1. Neo4j

Это хранилище данных, предназначенное для создания сложных взаимосвязей между частями данных в «графе», а затем перемещение этих данных очень сложными способами, используя длинные запросы.

2. MongoDB

Хранилище документов с ключом, основанное на объектах JSON. То есть для любого данного ключа есть большой «документ», состоящий из многих элементов. Доступ к документам осуществляется через ключи.

3. Cassandra

Похоже на MongoDB, поскольку так же является хранилищем для ключей, но вместо документов он хранит двоичные коды объектов.

#### 4. MySQL

Не смотря на то, что MySQL реляционная СУБД при использовании её с сервером OLAP Mondrian, можно составлять кубы данных и использовать их для анализа данных. [3]

В результате проведенного исследования можно сделать вывод, что система оценки состояния здоровья и подбора лекарственных средств должна отвечать следующим требованиям:

- предоставление пользователю результатов анализа за приемлемое время (обычно не более 5 с), пусть даже ценой менее детального анализа
- возможность осуществления любого логического и статистического анализа, характерного для данного приложения, и его сохранения в доступном для конечного пользователя виде;
- многомерное концептуальное представление данных, включая полную поддержку для иерархий и множественных иерархий (это – ключевое требование OLAP);

А так же, проектируемая система будет использовать СУБД MySQL с сервером OLAP Mondrian.

#### Литература

1. OLAP. Режим доступа: <https://ru.wikipedia.org/wiki/OLAP>
2. MySQL. Режим доступа: <https://ru.wikipedia.org/wiki/MySQL>
3. Mondrian(сервер OLAP). Режим доступа: <https://ru.wikipedia.org/wiki/Mondrian>
4. Введение в многомерный анализ. Режим доступа: <https://habrahabr.ru>

Осипов М.Г.

*Научный руководитель: ст. преп. каф. САПР Я.Ю. Кульков  
Муromский институт (филиал) федерального государственного образовательного  
учреждения высшего образования «Владимирский государственный университет  
имени Александра Григорьевича и Николая Григорьевича Столетовых»  
602264, г. Муром, Владимирская обл., ул. Орловская, 23  
E-mail: 89040307970@mail.ru*

### **Программа построения и диагностики компьютерной сети**

В настоящее время повсеместно распространены компьютерные сети, особенно на предприятии АО «МЗ РИП», где количество различных устройств, которые подключены к сети и должны быть отображены на карте превышает тысячу элементов. В связи с этим возникает необходимость в проектировании сети.

Проектирование представляет собой процесс определения архитектуры, компонентов, интерфейсов и других характеристик системы или её части (ISO 24765). Результатом проектирования является проект — целостная совокупность моделей, свойств или характеристик, описанных в форме, пригодной для реализации системы.

Проектирование системы направлено на представление системы, соответствующее предусмотренной цели, принципам и замыслам; оно включает оценку и принятие решений по выбору таких компонентов системы, которые отвечают её архитектуре и укладываются в предписанные ограничения [1].

В связи с этим возникает необходимость создания специализированной программы для проектирования компьютерной сети.

Такая программа дает возможность удобно проектировать компьютерную сеть со всеми особенностями строения сети предприятия, позволяет проводить манипуляции с информацией о сети, предоставлять информацию в удобном для просмотра виде. В программе реализовано разделение функционала для двух ролей – пользователь и администратор. Администратор может задавать права доступа для пользователей, а также задавать глобальные параметры программы.

Разработанная программа обладает следующими функциями:

- рисование и работа с геометрическими фигурами (линия, прямоугольник, многоугольник, круг);
- работа со зданиями, а именно преобразование геометрической фигуры в здание (кроме линий) с указанием числа этажей и наименования здания;
- рисование и добавление различных элементов карты и сети для определенного уровня (для главного плана или для определенного этажа в здании);
- фильтры, позволяющие отображать только нужные пользователю элементы;
- добавление сетевых устройств с использованием шаблонов, а также с заданием параметров их функционирования (параметрами являются тип устройства, количество портов, пропускная способность);
- сохранение и загрузка карты сети;
- ведение лога изменений для карты, показывающего какое изменение было произведено, кем и когда;
- отмена изменений, привязанная к логу;
- экспорт карты сети в различные форматы, а именно: изображение, PDF, список элементов карты с их параметрами;
- создание, удаление и редактирование заметки об элементе;
- поиск нужных зданий, помещений и элементов;
- проверка текущего состояния устройств (включено или выключено);

### **Литература**

1. Wikimedia Foundation // 2018. URL: <https://ru.wikipedia.org/wiki/Проектирование> (дата обращения: 01.04.2018).

Фомичев Ю.С.

*Научный руководитель: А.Д. Варламов*

**Муромский институт (филиал) федерального государственного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»  
602264, г. Муром, Владимирская обл., ул. Орловская, 23  
Россия, 602264, Владимирская область, г. Муром, ул. Орловская, д.23  
E-mail: foma532.yurok@yandex.ru**

Поиск сюжетных событий в литературных произведениях на основе анализа частот упоминания персонажей

В настоящее время автоматическая обработка текста – это стремительно развивающаяся область научных исследований, которая направлена на разработку алгоритмов и методов обработки больших объемов неструктурированной информации. Основные трудности, возникающие при решении задач обработки текстов, связаны с необходимостью работы с неструктурированными данными [3, 4]. Поэтому еще не удалось достичь универсального подхода к построению алгоритма, так как каждый конкретный алгоритм определяется строем языка.

На практике анализом больших текстов занимаются литературные критики и рецензенты. Статьи, написанные русскими и зарубежными критиками, продолжают и сегодня оказывать большое влияние на духовную и нравственную жизнь общества [2]. В процессе анализа произведений критикам необходимо не только прочесть большой объем текста, но и многократно возвращаться к его эпизодам для более детального их анализа, оценивать своего рода статистику произведения и так далее. Поэтому много времени ими тратится на поиск интересных их частей произведения.

Современным критикам можно существенно упростить задачу исследования произведений с помощью информационных систем, быстро и эффективно выполняющих поиск по тексту [1, 5], что позволит, в общем, и исключить рутинный поиск в тексте при работе рецензента, чтобы сосредоточиться на сути анализа и не отвлекаться по мелочам.

В процессе исследовательской работы было установлено, что большой информативностью обладает визуализация периодичности появления героев в произведении. Поэтому был разработан алгоритм визуализации активности героев в литературных текстах, который состоит из следующей последовательности операций:

1. Предварительная обработка произведения: разбиение текста на предложения, удаление лишних символов в начале и конце каждого предложения;
2. Поиск слов, начинающихся с заглавной буквы, кроме тех, что стоят в начале предложений, и оценка частот их встречаемости в тексте;
3. Отсевание редких слов из результатов поиска
4. Объединение однокоренных слов в списке найденных, и суммирование их частот;
5. Пересчет частот слов из списка найденных с учетом первых слов в предложениях;
6. Визуализация результатов поиска: построение графика частоты упоминания выбранного героя (или группы героев) во временном срезе.

Предложено использование этого алгоритма для быстрого поиска в литературных текстах сюжетных событий, сцен, актов, опираясь на участников этих событий – персонажей произведения.

В ходе проделанной работы был достигнут результат в виде функционирующей программы, которая облегчит литературным критикам и рецензентам анализ произведений и позволит быстро и продуктивно, перемещаясь в ту или иную часть произведения, достигать более точных и конкретных результатов в своей работе, затрачивая на это в разы меньше времени, чем при классическом поиске в произведении.

### Литература

1. Алексеев А.А., Катасёв А.С., Кириллов А.Е., Кирпичников А.П. Классификация текстовых документов на основе технологии text mining // Вестник Казанского технологического университета, 2016, Т19, №18, с. 116 – 119.
2. Канунова Е.Е. Вопросы автоматизации музейного дела // Алгоритмы, методы и системы обработки данных. 2014. № 4 (29). С. 72-76.
3. Ленкин А.В., Баженов Р.И. Исследование систем для Text Mining // Постулат. 2017. № 1 (15). С. 3.
4. Шарапова Е.В., Шарапов Р.В. Универсальная система проверки текстов на плагиат "автор.net" // Информатика и ее применения. 2012. Т. 6. № 3. С. 52-58.
5. Щербатов И.А., Беляев И.О. Многоагентная поисковая система: применение фибоначчиевых куч // Алгоритмы, методы и системы обработки данных. 2015. № 2 (31). С. 86-92.

Черняков А.А.

*Научный руководитель: доцент каф. САПР А.А. Быков  
Муромский институт (филиал) федерального государственного образовательного  
учреждения высшего образования «Владимирский государственный университет  
имени Александра Григорьевича и Николая Григорьевича Столетовых»  
602264, г. Муром, Владимирская обл., ул. Орловская, 23  
E-mail: damian25091995@gmail.com*

### **Проектирование и разработка автоматизированной информационной системы (АИС) учета научной работы студентов**

В современном мире наука стремительно развивается и активно занимает умы все большего числа людей. Появляются новые университеты, институты, лаборатории, развиваются молодые направления научной деятельности. С каждым годом все больше людей могут получать качественное высшее образование, реализовывать свои идеи.

Для того, чтобы студенты смогли не только получить определенную сумму знаний в соответствии с программой, но и научились самостоятельному поиску информации, совершенствованию знаний, которые они получили в ВУЗе, анализу и обобщению различной информации, должен проводиться комплекс мероприятий, называемый научно-исследовательской работой студентов (НИРС).

НИРС – неотъемлемая часть образовательного процесса, позволяющая студентам применять свои творческие возможности с первых дней обучения, которая является необходимым компонентом их профессиональной подготовки.

Для того, чтобы организовать хранение информации о научно-исследовательских работах студентов, необходима автоматизированная информационная система (АИС) учета НИРС. Такая система поможет организовать качественное и удобное хранение всей необходимой информации, позволит проводить манипуляции с этой информацией, составлять гибкие интерактивные отчеты, представлять всю актуальную информацию в удобном для просмотра виде.

Для выполнения большинства задач системы будет использована база данных, что позволит получить неоспоримые преимущества, такие как легкий поиск нужной записи, быстрая сортировка записей за счет того, что база данных хранит информацию достаточно эффективно, что позволяет минимизировать время доступа к данным.

Создание и управление базами данных обычно возможно с использованием систем управления базами данных (СУБД). СУБД – это комплекс программных средств, предназначенных для создания структуры базы данных, наполнения ее содержимым, редактирования как структуры, так и содержимого.

Для обеспечения взаимодействия между базой данных и приложением должны использоваться поставщики данных, которые инкапсулируют механизм работы с конкретной системой управления базами данных.

В ходе интервью с заказчиком было установлено, что система должна взаимодействовать с одной из нескольких СУБД, в зависимости от установок конкретного пользователя.

На текущий момент для учета НИРС студентов возможно использовать ряд программного обеспечения в комплексе. Например, для хранения данных, а также формирования отчетов возможно использовать MS Access, для построения гистограмм – MS Excel, для чтения отчетов – MS Word, для хранения резервных копий – OneDrive, Google Drive или Яндекс Диск.

Полученное в результате проектирования и разработки программное обеспечение позволит использовать все возможности, необходимые и обязательные для АИС учета научно-исследовательской работы студентов, в одном приложении. Данный программный продукт можно будет использовать преподавателям в ВУЗах для того, чтобы удобно и быстро работать с НИРС, получать доступ к необходимой информации, формировать удобные для чтения отчеты, собирать статистику по работам, обмениваться нужной информацией друг с другом.

Шекалин К.А.

*Научный руководитель: к.т.н., доцент каф. САПР А.А. Быков  
Муромский институт (филиал) федерального государственного образовательного  
учреждения высшего образования «Владимирский государственный университет  
имени Александра Григорьевича и Николая Григорьевича Столетовых»  
602264, г. Муром, Владимирская обл., ул. Орловская, 23  
E-mail: kir123455@rambler.ru*

### **Генерация баз данных на основе концептуальной модели**

Базы данных играют важную роль в развивающемся мире технологий. Всё, с чем мы каждый день взаимодействуем в жизни, зафиксировано в какой-нибудь базе. Работа с базами данных является важнейшим навыком в работе с компьютером, а специалисты данной области становятся всё более востребованными[1].

Для построения баз данных мало знать лишь информацию, которую нужно в нее занести. Чтобы построить эффективную и правильную базу данных нужно грамотно ее спроектировать, то есть создать модели базы данных. К моделям проектируемой базы данных относятся:

- Концептуальная модель;
- Логическая модель;
- Физическая модель[2].

Создание баз данных, при помощи моделей, сложно, имея подручные средства. Поэтому для решения рассмотренной задачи необходимо разработать программу, которая позволит создавать базы данных на основе моделей. В программе будут создаваться модели и на их основе программа автоматически создает базу данных, которую можно подключать в программы, которые работают с системой управления базами данных (СУБД).

Целью программы является облегчение создания баз данных, позволяя знать лишь минимальный набор сведений для ее проектирования.

В программе будет реализовано:

- Автоматизированное создание базы данных на основе физической, логической, концептуальной моделей;
- Создание объектов моделей с указанием необходимых атрибутов и связей;
- Приведение логической модели к нормальным формам;
- Сохранение результатов моделирования с возможностью экспорта в различные программы.

### **Литература**

1. Карпов А.П. Область применения баз данных: Материалы VIII Международной студенческой электронной научной конференции «Студенческий научный форум» [Электронный ресурс] URL: <http://www.scienceforum.ru/2016/1508/18214> (дата обращения: 31.03.2018)
2. Проектирование баз данных [Электронный ресурс] // Wikipedia.org: Сетевая энциклопедия. URL: [https://ru.wikipedia.org/Проектирование\\_баз\\_данных](https://ru.wikipedia.org/Проектирование_баз_данных) (дата обращения 31.01.18)