Федеральное государственное автономное образовательное учреждение высшего профессионального образования «Уральский федеральный университет имени первого Президента России Б.Н.Ельцина»

Кафедра «Теплофизика и информатика в металлургии»

Оценка проекта: \_\_\_\_\_\_\_\_\_\_\_\_\_

Члены комиссии:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

Подпись расшифровка подписи

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

Подпись расшифровка подписи

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

Подпись расшифровка подписи

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 201\_ г.

**Программная реализация модуля регистрации пользователей лабораторного стенда в лаборатории АСУТП с использованием технологии баз данных**

**КУРСОВАЯ РАБОТА**

**по дисциплине «Проектирование баз данных»**

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

230201.000.012 ПЗ

Руководитель

Доцент, к.т.н. В.В. Лавров

должность, звание подпись расшифровка подписи

Студент

Мт – 46052 Ю.А.Бородулин

номер группы подпись расшифровка подписи

Екатеринбург 2010

**Федеральное государственное автономное образовательное учреждение высшего профессионального образования «Уральский федеральный университет имени первого Президента России Б.Н.Ельцина»**

**Кафедра «Теплофизика и информатика в металлургии»**

УТВЕРЖДАЮ:

Зав. кафедрой\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (Н.А.Спирин)

« 15 » марта 2010 г.

**Задание № 2**

**на выполнение курсового проекта**

**по дисциплине «Проектирование баз данных»**

Студент группы **Мт-45052**. Специальность: 230201 – «Информационные системы и технологи»

Фамилия: **Бородулин**. Имя: **Юрий**. Отчество: **Александрович**.

Руководитель курсовой работы: к.т.н., доцент Лавров В.В. ([lavll2007@rambler.ru](mailto:lavll2007@rambler.ru))

Срок выполнения работы: с «15» марта 2010 г. по «16» мая 2010 г.

1. Тема курсовой работы: Программная реализация модуля регистрации пользователей лабораторного стенда в лаборатории АСУТП с использованием технологии баз данных.

2. Исходные данные к курсовому проекту: материалы предметной области, согласованные с представителем Заказчика.

3. Содержание курсовой работы

3.1. Пояснительная записка:

* титульный лист;
* бланк задания преподавателя;
* оглавление;
* введение;
* назначение, цели и задачи создания информационной системы;
* архитектура построения информационной системы;
* разработка функциональной модели системы;
* инфологическое и даталогическое моделирование базы данных;
* физическая реализация базы данных, автоматическая загрузка тестовых данных;
* блок-схема алгоритма обработки данных в системе;
* разработка приложения по сопровождению базы данных;
* разработка системы отображения отчетов;
* функциональные возможности информационной системы;
* заключение;
* список литературы;
* приложение (техническое задание на проект, подписанное преподавателем; руководство администратора; руководство пользователя информационной системы).

3.2. Компьютерные версии:

* функциональная модель (IDEF1X-диаграммы в пакете BPWin 7);
* инфологическая модель (ER-диаграммы в пакете ERWin 7);
* файл MS Office Excel для тестовой загрузки данных в базу;
* пакет Integration Services (IS) для автоматического переноса тестовых данных из MS Office Excel в MS SQL Server 2005;
* пакет Reporting Services (RS) для отображения в численном и графическом виде отчетных показателей на Web-сервере отчетов;
* backup-копия базы данных;
* прикладная программа для сопровождения системы (дистрибутив, исходники);
* внешний help-файл (откомпилированный, исходники).

4. Особые дополнительные требования

4.1. Функциональная модель

Выполняется по стандарту IDEF1X в пакете BPWin 7.

4.2. Концептуальная модель

Выполняется в виде ER-диаграмм в пакете ERWin 7.

4.3. Даталогическая модель

Генерируется из CASE-программы в целевую СУБД.

4.4. Пакет Integration Services (IS)

Пакет автоматической передачи тестовых данных из MS Office Excel в базу данных MS SQL Server 2005 должен быть реализован с использованием службы интеграции (Integration Services) сервера MS SQL Server 2005. Пакет IS разработать в среде Business Intelligence Development Studio, входящей в состав MS SQL Server 2005;

4.5. Пакет Reporting Services (RS)

Создание пакетов Reporting Services для отображения в численном и графическом виде отчетных показателей выполнить в среде Business Intelligence Development Studio, входящей в состав MS SQL Server 2005. Численное представление отчетных данных с использованием инструмента Reporting Services должно включать в себя использование сводных таблиц, интерактивных элементов (выпадающих списков, календарей, расширяемых областей отчета и пр.) для диалога с пользователем. Графическое представление отчетных данных с использованием инструмента Reporting Services должно включать в себя использование трендовых диаграмм (Chart), интерактивных элементов (выпадающих списков, календарей и пр.) для диалога с пользователем. На диаграмме предусмотреть возможность отображения нескольких линий трендов для сравнения динамики изменения отчетных показателей.

4.6. Обработка данных

Выполняется средствами прикладной программы и СУБД. В случае использования MS SQL Server 2005 обработку выполнить преимущественно с помощью хранимых процедур (stored procedures) и организации представлений (views). Взаимодействие прикладной программы и сервера организовать с помощью SQL-запросов и вызова хранимых процедур.

4.7. Прикладная программа

Разработка в среде программирования Visual Studio 2005 .NET; комментарии в исходном коде; контроль данных при заполнении форм; табличная и графическая интерпретации полученных результатов; контекстно-зависимая справка.

5. Календарный план выполнения проекта.

Продолжительность выполнения курсового проекта составляет 8 недель, основные фазы, этапы и стадии проектирования приведены в таблице.

Таблица – Основные фазы выполнения курсового проекта

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № п/п | Наименование элементов  проектной работы | Сроки | Примечания | Отметки о выполнении |
| 1 | Анализ предметной области, подготовка ТЗ | 15.03.2010 –  22.03.2010 | Сдача ТЗ в виде документа MS Word |  |
| 2 | Разработка проектных решений. Проектирование архитектуры системы, базы данных, программных модулей приложения | 23.03.2010–  05.04.2010 | Внешнее описание программного средства в виде документа MS Office Word. Инфологическая модель в виде ER-диаграмм в пакете ERWin |  |
| 3 | Создание файла тестовых данных MS Office Excel и пакета интеграции данных IS | 06.04.2010 –  12.04.2010 | Файл тестовых данных MS Office Excel и пакет интеграции данных IS |  |
| 4 | Создание пакета отображения отчетных данных RS на Web-сервере отчетов | 13.04.2010 –  19.04.2010 | Пакет отображения отчетных данных RS |  |
| 5 | Разработка функциональной модели программного средства | 20.04.2010 –  26.04.2010 | Сдача файла c IDEF-диаграммами в виде документа BPWin |  |
| 6 | Реализация базы данных и приложения. Кодирование программного средства. Тестирование и отладка системы. Создание дистрибутива | 27.04.2010 –  03.05.2010 | Сдача тестовой версии программного средства (альфа-релиз)  Backup-копия базы данных c тестовыми данными |  |
| 7 | Разработка файла контекстно-зависимой справки | 04.05.2010 –  10.05.2010 | Сдача руководства пользователя в виде chm-файла и pdf-документа |  |
| 8 | Подготовка пояснительной записки | 11.05.2010 –  15.05.2010 | Сдача ПЗ в виде документа MS Word |  |
| 9 | Сдача работы | 16.05.2010 | Все материалы по проекту |  |

6. Форма отчетности:

* заполненные электронные папки со всеми перечисленными материалами;
* пояснительная записка – твердая копия и электронный вариант.

7. Выполнение курсового проекта закончено «\_\_\_»\_\_\_\_\_\_\_\_2010 г.

Руководитель: доц., канд. техн. наук В.В.Лавров

Задание получил: «\_\_\_»\_\_\_\_\_\_\_\_2010 г. Ю.А.Бородулин

# Аннотация

Пояснительная записка изложена на 35 листах и содержит 4 таблицы, 22 рисунка и 2 приложения.

В данной пояснительной записке к курсовой работе приведено описание процессов разработки программного продукта, реализованного на языке C# в Visual Studiо 2005 с использованием СУБД MS SQL Server 2005, посвященного теме автоматизации сбора информации при проведении лабораторных работ.

Рассмотрены основные этапы разработки программного обеспечения: постановка задачи, разработка технического задания, алгоритмического обеспечения, проектирование и реализация программного средства, создание справочной документации, подготовка дистрибутива.

Проект представляет собой программу, позволяющую автоматизировать процессы работы, связанной с проведением лабораторных работ. В разделах пояснительной записки дается постановка задачи, описание программных средств, используемых для создания проекта, подробное описание разработанного программного продукта.

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 6](#_Toc279084154)

[1. ПОСТАНОВКА ЗАДАЧИ 7](#_Toc279084155)

[1.1 Организация работы пользователей с созданным АРМ 7](#_Toc279084156)

[1.2 Архитектура построения информационной системы 7](#_Toc279084157)

[1.3 Разработка функциональной модели 9](#_Toc279084158)

[2. РАЗРАБОТКА СТРУКТУРЫ ХРАНЕНИЯ ДАННЫХ 11](#_Toc279084159)

[2.1 Разработка инфологической модели 11](#_Toc279084160)

[2.2 Разработка даталогической модели 12](#_Toc279084161)

[2.3 Обработка данных 13](#_Toc279084162)

[3. РАЗРАБОТКА ПРИЛОЖЕНИЯ БАЗЫ ДАННЫХ 15](#_Toc279084163)

[3.1 Блок-схема алгоритма обработки данных 15](#_Toc279084164)

[3.2 Физическая реализация базы данных 16](#_Toc279084165)

[3.3 Разработка прикладной программы 17](#_Toc279084166)

[3.4 Создание контекстно-зависимой справки 19](#_Toc279084167)

[3.5 Создание дистрибутива 19](#_Toc279084168)

[4. ХАРАКТЕРИСТИКА СИСТЕМЫ 21](#_Toc279084169)

[4.1 Подготовка базы данных 21](#_Toc279084170)

[4.2 Установка и настройка программного средства 22](#_Toc279084171)

[ЗАКЛЮЧЕНИЕ 23](#_Toc279084172)

[СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ 24](#_Toc279084173)

[ПРИЛОЖЕНИЕ А 25](#_Toc279084174)

[ПРИЛОЖЕНИЕ Б 34](#_Toc279084199)

# ВВЕДЕНИЕ

В наши дни любая организация просто не может представить свою работу без решений в сфере ИТ, которые упрощают и ускоряют работу.

Любой рутинный процесс может быть автоматизирован. В наши дни любая уважающая себя организация ведет базу по своей работе в электронном виде, так как это наиболее простой и удобный способ хранения любой информации. Ну а тем более автоматизация не может обойти стороной такой объект, как лабораторный стенд в лаборатории АСУТП на нашей кафедре. Поэтому в данной курсовой работе я наглядно решил продемонстрировать работу базы данных и программного интерфейса для использования при выполнении лабораторных работ.

Целью данной курсовой работы является создание базы данных и программного интерфейса для сбора информации о проводимых лабораторных работах.

Для достижения этой цели, необходимо выполнить следующие задачи:

* создание модуля аутентификации и авторизации пользователей;
* создание модуля отображение и корректировки данных из базы;
* защита от некорректного ввода данных;
* создание отчетной документации;
* создание файла справки.

# 1. ПОСТАНОВКА ЗАДАЧИ

## 1.1 Организация работы пользователей с созданным АРМ

Организацию работы пользователей планируется проводить согласно рисунку 1.1, на котором отражено взаимодействие пользователя с программным продуктом.

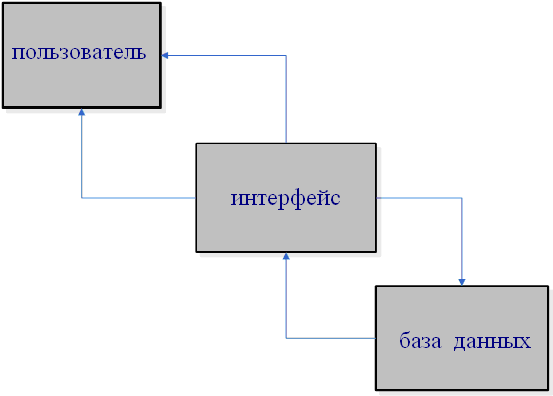


Рисунок 1.1 – Схема взаимодействия пользователя с созданным АРМ

Рассмотрим задачу более детально. Для того чтобы начать работу с базой данных, пользователь должен авторизоваться, чтобы подтвердить свои полномочия для подобной работы.

База данных лабораторного стенда хранит в себе информацию о проведенных лабораторных работах, а именно: какой студент проводил лабораторную, под чьим руководством и какие были получены результаты.

Все эти данные хранятся в базе, в которую вносятся ответственным лицом – оператором АРМ. В базе эти данные доступны также для корректировки и удаления. Связь с базой осуществляется при помощи интуитивно понятного программного интерфейса.

## 1.2 Архитектура построения информационной системы

Для выполнения групп функций программы создан свой модуль, который взаимодействует с другими модулями через определенный интерфейс. Архитектура системы показана на рисунке 1.2.

Пользователь программы взаимодействует с ней через графические объекты, которые функционируют под управлением одного из модулей.

Модуль загрузки обеспечивает взаимодействие с базой данных, расположенной на сервере базы данных, открытие и чтение файлов с исходными данными или загрузку данных, определенных по умолчанию, а также осуществляет возможность сохранения файла параметров.

Рисунок 1.2 – Архитектура информационной системы

Для того, чтобы начать работу с приложением, пользователю необходимо пройти процесс авторизации, при котором будет указываться имя пользователя, пароль, сервер и название базы данных, которая хранит данные.

После успешно пройденного процесса авторизации пользователя, осуществляется загрузка данных из БД из файла формат \*.CSV.

Пользователь данного приложения имеет возможность вносить новые данные в базу данных, корректировать и удалять их. Поскольку все эти данные хранятся в БД, то существует возможность просматривать все внесенные заказы.

После выполнения всех необходимых операций с пользовательскими данными, существует возможность создания отчета с помощью Reporting Service, также предусмотрен экспорт этих данных в такие общедоступные форматы, как PDF и MS Excel (\*.xls).

Хранение данных организовано при помощи СУБД MS SQL Server 2005. Непосредственная организация данных будет описана далее.

## 1.3 Разработка функциональной модели

Функциональная модель программного средства ПО «Лабораторный стенд» выполнена по стандарту IDEF0 с использованием CASE-средства BPWin, фрагмент модели изображен на рисунках 1.3, 1.4.

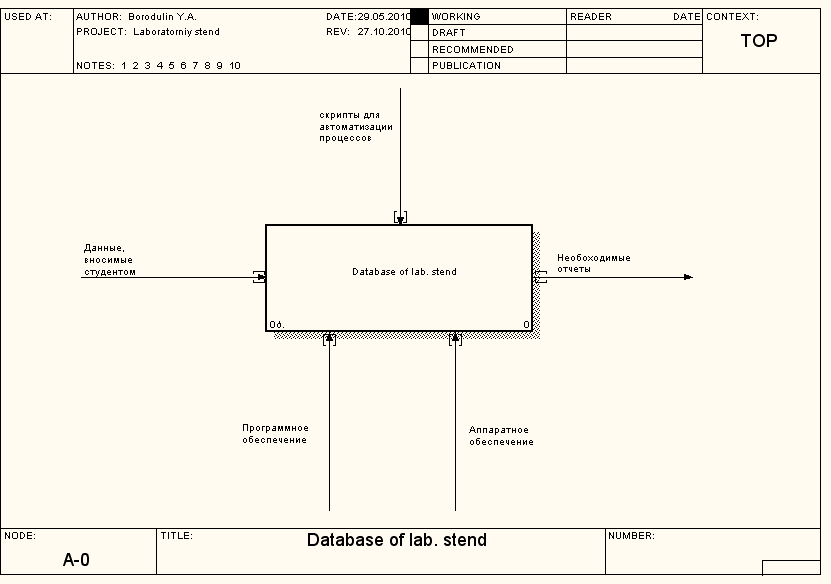


Рисунок 1.3 – Функциональная модель программного средства (главная форма)

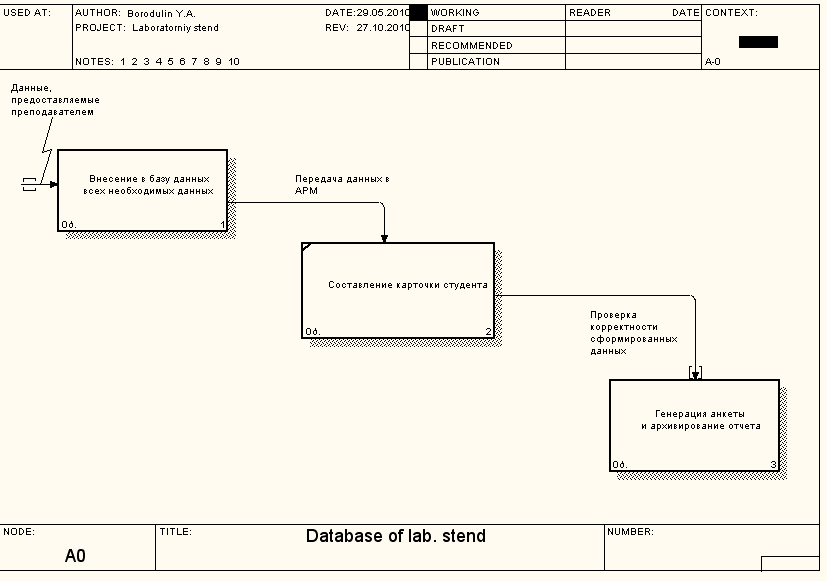


Рисунок 1.4 – Функциональная модель программного средства (развернутая форма)

Программное средство должно обеспечивать пять основных функций:

* аутентификация и авторизация пользователя;
* отображение и корректировка данных в базе данных;
* ввод исходных данных;
* создание отчета;
* экспорт отчета в известные форматы (.pdf, .xls).

Программное средство должно производить аутентификацию и авторизацию пользователей. Это необходимо для обеспечения целостности и правильности хранения данных о пользователях беспроводной сети.

Поскольку база находится в режиме постоянного обновления, то требуется корректировать данные, хранящиеся в базе.

# 2. РАЗРАБОТКА СТРУКТУРЫ ХРАНЕНИЯ ДАННЫХ

## 2.1 Разработка инфологической модели

Инфологическая модель выполнена в виде ER-диаграмм с использованием CASE-пакета в программе ERWin. Инфологическая модель состоит из четырех таблиц, представленных на рисунке 2.1.

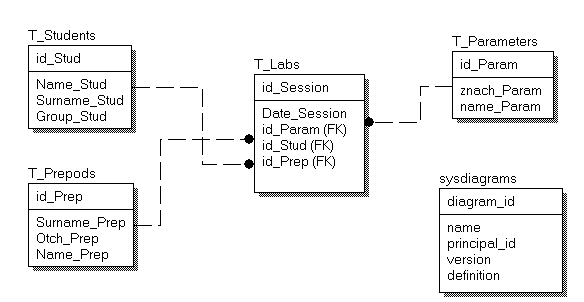


Рисунок 2.1 – Инфологическая модель базы данных

В таблицах 2.1-2.5 представлена спецификация инфологической модели проектируемой системы.

|  |  |  |
| --- | --- | --- |
| № поля | Название поля | Тип поля |
| 1 | id студента | (key)integer |
| 2 | Имя студента | Nvarchar(50) |
| 3 | Фамилия студента | Nvarchar(50) |
| 4 | Группа | Nvarchar(50) |

Таблица 2.1 – T\_ Students

Таблица T\_ Students содержит в себе данные о студентах. Вся информация о лабораторных работах вносятся в таблицу T\_ Labs.

|  |  |  |
| --- | --- | --- |
| № поля | Название поля | Тип поля |
| 1 | id сессии | (key)integer |
| 2 | Дата сессии | smalldatetime |
| 3 | id параметра | integer |
| 4 | id студента | integer |
| 5 | id преподавателя | integer |

Таблица 2.2 – T\_ Labs

Лабораторные работы проводятся под руководством одного из преподавателей. База должна быть наполнена данными о них (табл. 2.3):

|  |  |  |
| --- | --- | --- |
| № поля | Название поля | Тип поля |
| 1 | id преподавателя | (key)integer |
| 2 | Фамилия преподавателя | Nvarchar (50) |
| 3 | Имя преподавателя | Nvarchar (50) |
| 4 | Отчество преподавателя | Nvarchar (50) |

Таблица 2.3 – T\_Prepods

При проведении лабораторных работ результатом являются числовые значения измеряемых технологических параметров. Данные по ним содержатся в таблице T\_ Parameters (табл. 2.4).

|  |  |  |
| --- | --- | --- |
| № поля | Название поля | Тип поля |
| 1 | id параметра | (key)integer |
| 2 | Значение параметра | Nvarchar(50) |
| 3 | Наименование параметра | Nvarchar(50) |

Таблица 2.4 – T\_ Parameters

## 2.2 Разработка даталогической модели

Даталогическая модель включает в себя схему данных БД в рамках выбранной СУБД MS SQL Server 2005. Даталогическая модель представлена на рисунках 2.2-2.5.

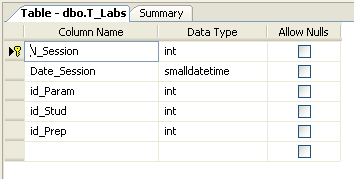


Рисунок 2.2 – Таблица dbo.T\_Labs

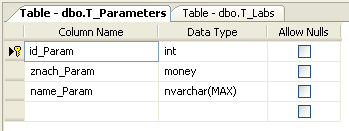


Рисунок 2.3 – Таблица dbo.T\_Parameters

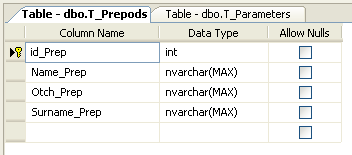


Рисунок 2.4 – Таблица dbo.T\_Prepods

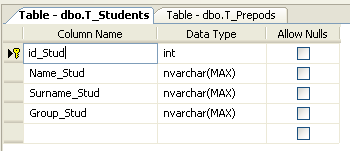


Рисунок 2.5 – Таблица dbo.T\_Students

Для реализации обмена данными между программой и базой данных были написаны соответствующие хранимые процедуры, которые позволяют группировать данные из разных таблиц.

## 2.3 Обработка данных

Любое взаимодействие с БД происходит с помощью хранимых процедур. Можно выделить четыре основных типов хранимых процедур, используемых в программе:

* для вставки данных;
* для удаления данных;
* для корректировки данных;
* для выборки данных;

Для реализации задач курсовой работы написано 4 хранимые процедуры. Ниже приведены 2 листинга для процедур dbo.ex\_DEL\_Students и dbo.ex\_INS\_Students (рисунки 2.6 и 2.7 соответственно).

|  |
| --- |
| set ANSI\_NULLS ON  set QUOTED\_IDENTIFIER ON  go  /\*-----------------------------------------------------------    Пример обработки базы данных в среде Visual Studio 2005 .NET"  Входные данные:  @id - идентификатор пользователя  Пример вызова:  EXEC dbo.ex\_DEL\_Students @id\_Stud = 5  \*/-----------------------------------------------------------  ALTER PROC [dbo].[ex\_DEL\_Students]  @id\_Stud int  AS  SET NOCOUNT OFF -- передавать сообщения о количестве обработанных записей  SET ANSI\_WARNINGS OFF -- отключить вывод предупреждений  -- Удалить запись  DELETE FROM dbo.T\_ Students  WHERE id\_Stud = @id\_Cust  SET NOCOUNT ON -- не передавать сообщения о количестве обработанных записей  -- Прочитать запись  SELECT id\_Stud,  Name\_Stud,  Surname\_Stud,  Group\_Stud    FROM dbo.T\_Students  WHERE id\_Stud = @id\_Stud  SET ANSI\_WARNINGS ON -- включить вывод предупреждений |

Рисунок 2.6 – Листинг процедуры dbo.ex\_DEL\_Students

|  |
| --- |
| set ANSI\_NULLS ON  set QUOTED\_IDENTIFIER ON  go  ALTER PROC [dbo].[ex\_INS\_Students]  @id\_Stud int,  @Name\_Stud nvarchar(max),  @Surname\_Stud nvarchar(max),  @Group\_Stud nvarchar(max)  as  SET NOCOUNT OFF -- передавать сообщения о количестве обработанных записей  -- Проверить существование строки: если не существует, то вставить  IF NOT EXISTS ( SELECT 1 FROM dbo.T\_Students WHERE id\_Stud = @id\_Stud )  BEGIN  INSERT INTO dbo.T\_Students (  id\_Stud,  Name\_Stud,  Surname\_Stud,  Group\_Stud  )  VALUES (  @id\_Stud,  @Name\_Stud,  @Surname\_Stud,  @Group\_Stud  )  SET NOCOUNT ON -- отключить сообщения о количестве обработанных записей  -- Прочитать новую запись  SELECT id\_Stud,  Name\_Stud,  Surname\_Stud,  Group\_Stud  FROM dbo.T\_Students  WHERE id\_Stud = @id\_Stud  END  ELSE  SET NOCOUNT ON |

Рисунок 2.7 – Листинг процедуры dbo.ex\_INS\_Students

# 3. РАЗРАБОТКА ПРИЛОЖЕНИЯ БАЗЫ ДАННЫХ

## 3.1 Блок-схема алгоритма обработки данных

Схема работы пользователя с программой представлена на рисунке 3.1.



Рисунок 3.1 – Алгоритм работы созданного приложения

Рассмотрим алгоритм более подробно. На первоначальном этапе, для входа в приложение пользователю предлагается авторизоваться, введя свои персональные данные и указав необходимые реквизиты для входа.

Пройдя процесс авторизации, пользователь может приступить к работе. Авторизованный пользователь может вносить в базу новые данные, редактировать или удалять имеющиеся. Для формирования отчетной деятельности предусмотрен фильтр. Сформированный отчет может быть экспортирован в удобный формат файлов.

## 3.2 Физическая реализация базы данных

В качестве СУБД была выбрана MS SQL Server 2005. SQL Server 2005 – это реляционная СУБД, которая использует язык Transact-SQL для пересылки сообщений между компьютером клиента и компьютером, на котором работает SQL Server 2005 [2].

В данной курсовой работе была создана база данных (рис 3.2), включающая в себя:

* 4 таблицы (dbo.T\_Students, dbo.T\_Labs, dbo.T\_Prepods, dbo.T\_Parameters);
* 1 диаграмма взаимодействия (dbo.Diagram);
* 16 хранимых процедур.

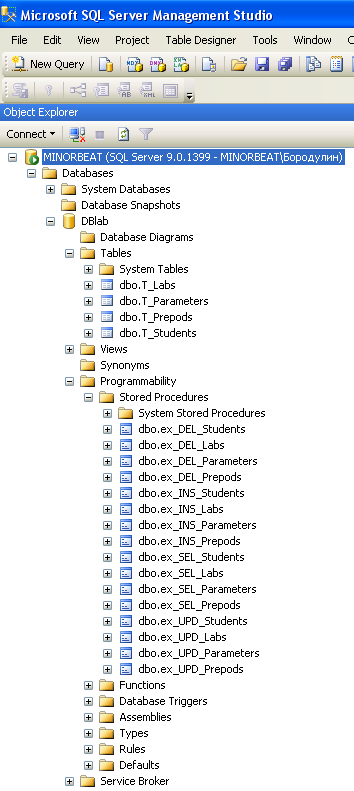


Рисунок 3.2 – Внешний вид базы данных в SQL Server 2005

## 3.3 Разработка прикладной программы

Интерфейс программного средства «ПО Лабораторный стенд» представляет собой совокупность взаимосвязанных окон. Взаимодействие пользователя и программы осуществляется при помощи текстовых полей, кнопок и других стандартных объектов Windows. Проектируемое программное средство создано в среде MS Visual Studio 2005.NET на языке C#.

Первоначально открывается форма регистрации пользователя, представленная на рисунке 3.3.

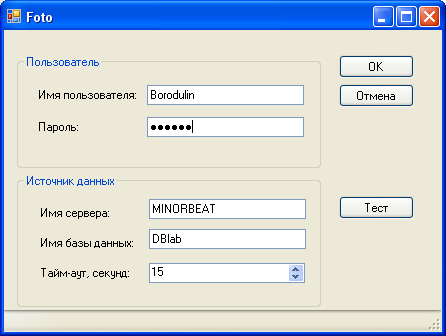


Рисунок 3.3 – Форма регистрации пользователя

В случае правильного ввода данных, при нажатии на кнопку «Тест», появится сообщение «Success», говорящее об успешно завершенном процессе авторизации пользователя. В случае же ошибочного ввода данных, появится сообщение, свидетельствующее о том, что данные были введены некорректно (рис 3.4):



Рисунок 3.4 – Сообщение, говорящее о некорректности введенных данных

При нажатии на кнопку Тест происходит проверка возможности соединения с базой данных. При нажатии на кнопку ОК происходит соединение с базой данных и открывается главная форма проекта (рис 3.5), в которой и будет происходить работа с данными.

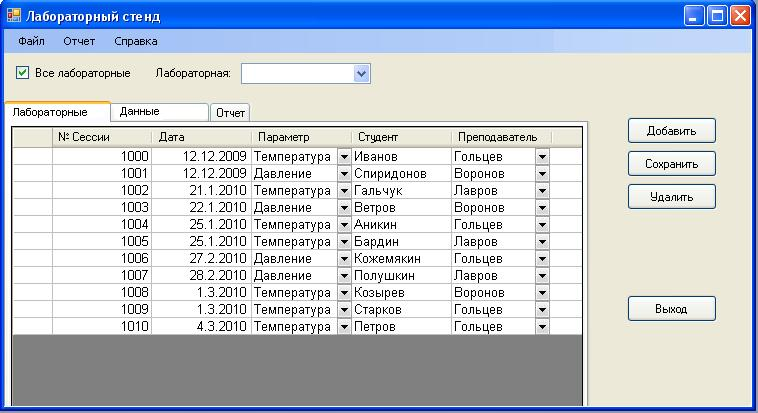


Рисунок 3.5 – Главная форма

На данном рисунке представлена главная форма работы с данными. Оператор АРМ имеет возможность добавлять, удалять или изменять существующие данные. Для того, чтобы просматривать имеющиеся заказы, предусмотрен фильтр по идентификационному номеру сессии выполнения лабораторной работы.

Данные, отфильтрованные по идентификационному номеру не доступны для редактирования в данном режиме.

После проведения всех необходимых действий, пользователь имеет возможность сгенерировать отчет с последующим экспорт в формат \*.xls или \*.pdf.

Для того, чтобы просмотреть данные о создатели данного приложения, предусмотрена форма «О программе» (рис 3.6):

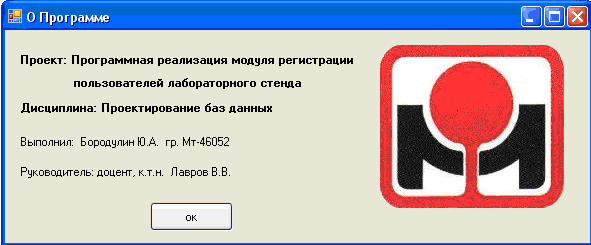


Рисунок 3.6 – О программе

На данной форме можно просмотреть информацию о создателях программы, теме проекта и т.д.

## 3.4 Создание контекстно-зависимой справки

В рамках проекта создана контекстно-зависимая справка в пакете Help&Doc, в которой отражены все возможности созданного программного продукта и включены указания по использованию программного средства.

Файл справки представляет собой скомпилированный HTML- файл. Также файл справки представлен в формате \*.hlp, \*.pdf.

На рисунке 3.7 представлено изображения файла справки в формате HTML.

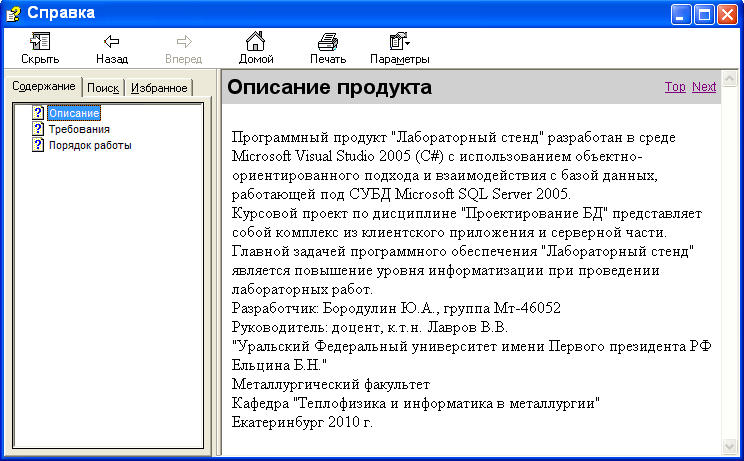


Рисунок 3.7 – Контекстно-зависимая справка

Справка вызывается путем нажатия меню «Помощь», затем выбора меню «Справка».

## 3.5 Создание дистрибутива

Для установки файлов программы на компьютер пользователя был создан дистрибутив. Дистрибутив создан с помощью MS Visual Studio 2005.NET .

С помощью мастера была определена вся необходимая информация для сценария создания установочного файла.

В свойствах были установлены следующие значения:

Author: Бородулин Ю.А.;

Description: АРМ для лабораторного стенда;

Localization: Russian;

Product name: ПО Лабораторный стенд;

Путь по умолчанию: C:\Program Files\Lab\_stend\;

Для построения дистрибутива следует нажать Build-> Build Setup.

В результате создается папка Setup, в которой находится дистрибутив Setup->Release-> Setup.exe.

Процесс установки дистрибутива представлен на рисунке 3.8.

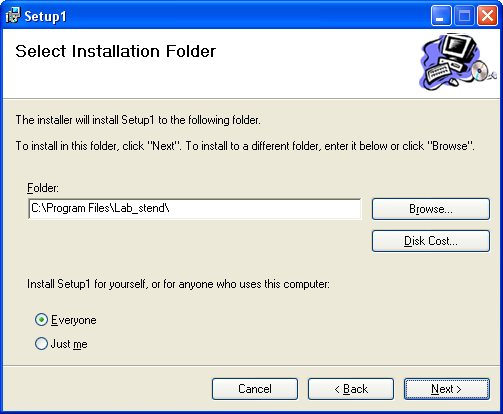


Рисунок 3.8 – Установка дистрибутива

# 4. ХАРАКТЕРИСТИКА СИСТЕМЫ

## 4.1 Подготовка базы данных

Для правильного функционирования системы необходимо установить на компьютере сервер БД MS SQL Server 2005. Для многопользовательской работы системы необходимо установить в свойствах сервера смешенный режим аутентификации. Необходимо создать пользователей на сервере (Enterprise Manager \ Security \ Logins \ New login).

Этапы подготовки БД:

1. запустить сервер MS SQL Server 2005, выбрать имя сервера (рисунок 4.1):

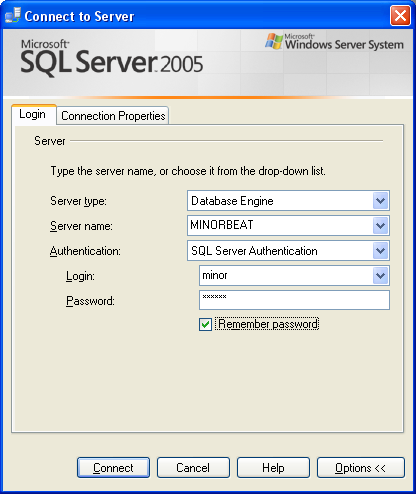


Рисунок 4.1 – Запуск сервера

1. В Enterprise Manager создать базу данных с именем DBlab (рисунок 4.2):

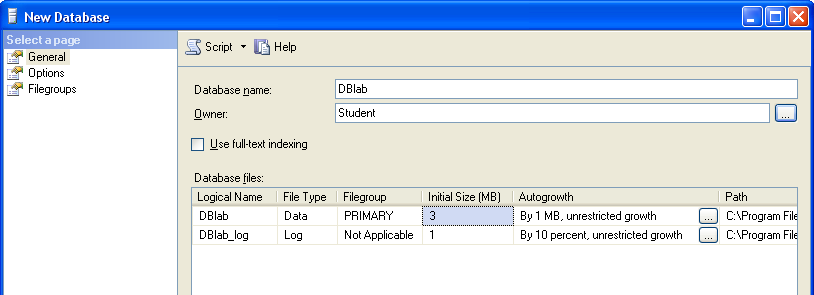


Рисунок 4.2 – Создание базы данных

1. Распаковать копию БД, указав место размещения копии БД (рисунок 4.3):

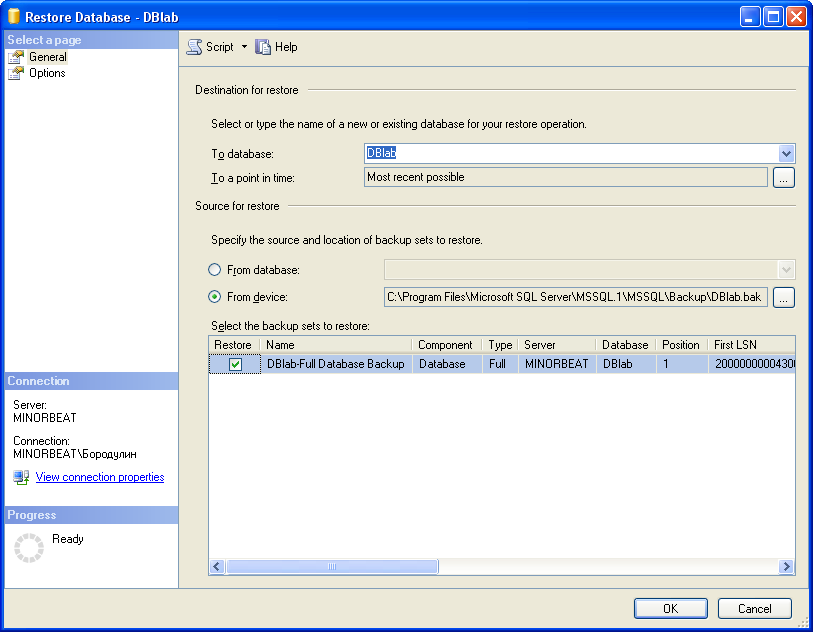


Рисунок 4.3 – Выполнение процедуры Restore

После этого база данных готова к использованию.

## 4.2 Установка и настройка программного средства

Чтобы использовать программу пользователь должен установить ее на свой компьютер. Для этого пользователю следует запустить на выполнение установочный файл программы setup.exe.

После запуска файла появится окно, в котором нужно указать полный путь к папке, куда следует установить файлы программы. По умолчанию файлы скопируются в папку «C:\Program Files\Setup». После этого нужно нажать кнопку «Начать» и тогда начнется копирование файлов в указанную папку. Если же пользователь передумал устанавливать программу, ему следует нажать кнопку «Выход».

После завершения копирования файлов появится окно с уведомлением об окончании установки.

По завершении установки на рабочем столе компьютера пользователя появится ярлык программы для ее запуска. В «Панели управления» в списке «Установка и удаление программ» появится наименование данной программы: «Лабораторный стенд».

После установки для данной программы не требуется дополнительная настройка.

# ЗАКЛЮЧЕНИЕ

В процессе проектирования и создания программного средства «ПО Лабораторный стенд» было создано следующее:

* функциональная схема проектируемой системы;
* инфологическая и даталогическая модели;
* генерация базы данных в СУБД MS SQL Server 2005;
* программное средство для работы с базой данных;
* справочная система программы;

Разработанное программное средство отвечает всем задачам, определенным в начале проектирования и обеспечивает заданную функциональность.

# СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. <http://ru.wikipedia.org>
2. SQL Server 2000. Ученый курс MCAD/MCSE, MCDBA. [Текст]: [пер. с англ. 2-е изд.] / Microsoft Corporation. – М.: Издательско-торговый дом «Русская Редакция», 2003. – 512 с.
3. Артемов Д.В. Microsoft SQL Server 2000. Новейшие технологии [Текст] / Д.В.Артемов. – М.: Издательско-торговый дом «Русская редакция», 2001. – 576 с.
4. Гамма Э. Приемы объектно-ориентированного проектирования. Паттерны проектирования [Текст]: [пер. с англ.] / Э.Гамма, Р.Хелм. – СПб.: Питер, 2007. – 368 с.
5. Дейт К.Дж. Введение в системы баз данных. [Текст]: [пер. с англ., 8-е изд.] / К.Дж.Дейт. – М.: Издательский дом «Вильямс», 2005. – 1328 с.
6. Карпова Т.С. Базы данных: модели, разработка, реализация [Текст] / Т.С.Карпова. – СПб.: Питер, 2001. – 304 c.
7. Конноли Т. Базы данных. Проектирование, реализация и сопровождение. Теория и практика. [Текст]: [пер. с англ., 3-е изд.] / Т.Конноли, К.Бегг. – М.: Издательский дом «Вильямс», 2003. – 1440 с.
8. Лошкарев Н.Б. Указания к оформлению дипломных и курсовых проектов и работ: Методические указания / Н.Б. Лошкарев, А.Н. Лошкарев, Л.А. Зайнуллин – Екатеринбург: ГОУ ВПО УГТУ–УПИ, 2007. – С.49. Прил. 9
9. Плещев В.В. Разработка и стандартизация программных средств и информационных технологий. CASE средства BPwin, ERwin с примерами и упражнениями: Учебное пособие [Текст] / В.В.Плещев. – Екатеринбург: изд. Уральского государственного экономического университета, 2003. – 208 с.
10. Фаронов В.В. Программирование баз данных в Delphi 7. Учебный курс [Текст] / В.В.Фаронов. – СПб.: Питер, 2004. – 464 с.
11. Фаронов В.В. Система программирования Delphi [Текст] / В.В.Фаронов. – СПб.: БХВ-Петербург, 2003. – 912 с.
12. Фаулер М. Архитектура корпоративных программных приложений [Текст]: [пер. с англ.] / М.Фаулер. – М.: Вильямс, 2006. – 544 с.
13. ГОСТ 24104–85 ЕСС АСУ «Автоматизированные системы управления. Общие требования»;
14. ГОСТ 24601–86 ЕСС АСУ «Автоматизированные системы управления. Стадии создания».
15. ГОСТ 34.201–89 «Информационная технология. Комплекс стандартов на автоматизированные системы. Виды, комплектность и обозначение документов при создании автоматизированных систем»;
16. ГОСТ 34.602–89 «Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы».

# ПРИЛОЖЕНИЕ А «Техническое задание на выполнение работы»

**Федеральное государственное автономное образовательное учреждение высшего профессионального образования**

**«Уральский федеральный университет имени Первого Президента Б.Н. Ельцина».**

**Кафедра «Теплофизика и информатика в металлургии»**

|  |
| --- |
| УТВЕРЖДАЮ: |
| Зав. кафедрой «Теплофизика  и информатика в металлургии»  Н.А. Спирин |
| «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_2010 г. |

**Программная реализация модуля регистрации пользователей лабораторного стенда в лаборатории АСУТП с использованием технологии баз данных.**

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ**

на выполнение курсовой работы

по дисциплине «Проектирование баз данных»

|  |  |  |
| --- | --- | --- |
| **от Заказчика** |  | **от Исполнителя** |
| Доцент кафедры ТИМ  В.В.Лавров |  | Студент группы МТ-46052  Ю.А.Бородулин |
| «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_2010 г. |  | «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_2010 г. |

Екатеринбург

2010

**Перечень использованных сокращений**

|  |  |
| --- | --- |
| ПО | Программное обеспечение |
| ПП | Программный продукт |
| ТИМ | Теплофизика и информатика в металлургии |
| АСУ | Автоматизированные системы управления |
| ОС | Операционная система |
| ЭТ | Электронные таблицы |
| ИД | Исходные данные |
| ООП | Объектно-ориентированное программирование |
| ЭВ | Электронный вид |
| ПЗ | Пояснительная записка |
| БД | База данных |

# 1. ОБЩИЕ СВЕДЕНИЯ

## 1.1. Наименование программного обеспечения

Программная реализация модуля регистрации пользователей лабораторного стенда в лаборатории АСУТП с использованием технологии баз данных.. Рабочее название проекта – ПО «Лабораторный стенд».

## 1.2. Шифр разработки

Шифр разработки 230201.000.012

## 1.3. Основание для выполнения работы

Учебный план специальности 230201 – «Информационные системы и технологии».

## 1.4. Стороны, участвующие в создании программного обеспечения

1.4.1. Заказчик – кафедра «Теплофизика и информатика в металлургии» ГОУ ВПО «УГТУ–УПИ» г. Екатеринбург. От имени Заказчика – доцент кафедры ТИМ, к.т.н. Лавров Владислав Васильевич. E-mail: [lavll2007@rambler.ru](mailto:lavll2007@rambler.ru).

1.4.2. Разработчик ПП (в дальнейшем – Исполнитель) – кафедра «Теплофизика и информатика в металлургии» ФГАОУ ВПО «УрФУ имени первого Президента России Б.Н.Ельцина» г.Екатеринбург. От имени Исполнителя – Бородулин Юрий Александрович, студент специальности 230201 «Информационные системы и технологии». E-mail: [u.borodulin06@net-ustu.ru](mailto:u.borodulin06@net-ustu.ru).

## 1.5. Исходные данные для разработки

Справочные материалы, согласованные с представителями Заказчика, а так же бланк задания, предоставленный Заказчиком.

## 1.6. Основание для разработки ПО «Лабораторный стенд»

ПО разрабатывается на основании следующего фактора – Заказчику требуется создание информационной системы на основе базы данных, используя СУБД MS SQL Server, а также создание программного интерфейса для обеспечения наибольшего комфорта при работе пользователей с созданной БД.

## 1.7. Плановые сроки начала и окончания работ по созданию программного средства

Срок начала разработки ПП «18» марта 2010 г, срок окончания работ по проекту «20» мая 2010 г. Продолжительность разработки программного средства и сдачи его в эксплуатацию 10 (десять) недель.

Последовательность и сроки реализации стадий и этапов работ, виды и комплектность документов, передаваемых Заказчику, определяются согласно табл. 1.1.

Таблица 1.1 Основные этапы разработки и сроки их выполнения

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Наименование элементов  проектной работы | Сроки | Примечания |
| 1 | Анализ предметной области, подготовка ТЗ | 15.03.2010 –  22.03.2010 | Сдача ТЗ в виде документа MS Word |
| 2 | Разработка проектных решений. Проектирование архитектуры системы, реализация базы данных, программных модулей приложения | 23.03.2010–  05.04.2010 | Внешнее описание программного средства в виде документа MS Office Word. Инфологическая модель в виде ER-диаграмм в пакете ERWin |
| 3 | Создание файла тестовых данных MS Office Excel и пакета интеграции данных IS | 06.04.2010 –  12.04.2010 | Файл тестовых данных MS Office Excel и пакет интеграции данных IS |
| 4 | Создание пакета отображения отчетных данных RS на Web-сервере отчетов | 13.04.2010 –  19.04.2010 | Пакет отображения отчетных данных RS |
| 5 | Разработка функциональной модели программного средства | 20.04.2010 –  26.04.2010 | Сдача файла c IDEF-диаграммами в виде документа BPWin |

# 2. НАЗНАЧЕНИЕ ПО И ЦЕЛИ ЕГО РАЗРАБОТКИ

## 2.1. Назначение программного средства

ПО «Лабораторный стенд» предназначено для автоматизации процессов работы, связанной с проведением лабораторных работ. Разработка среды основана на создании:

* базы данных студентов, результатов работ и руководителей работы;
* программного интерфейса для обеспечения взаимодействия пользователей с данной БД.

## 2.2. Цели создания

Основными целями разработки ПО «Лабораторный стенд» являются:

* создание условий для использования БД для проведения лабораторных работ и сохранения результатов;
* создание инструмента управления данной БД.

# 3. ХАРАКТЕРИСТИКА ОБЪЕКТА ИНФОРМАТИЗАЦИИ

Объект информатизации представляет собой взаимодействие пользователя с БД, в которой отражена основная информация о студентах и результатах проведенных лабораторных работ.

Пользователями данной БД являются студенты, выполняющие работы, а также преподаватели.

# 4. ТРЕБОВАНИЯ К ПРОГРАММНОМУ ОБЕСПЕЧЕНИЮ

## 4.1 Требования к ПО в целом

### 4.1.1 Требования к структуре и функционированию

1. БД должна включать основные таблицы: **Т\_Students, T\_Prepods, T\_Labs, T\_ Parameters.**

|  |  |  |
| --- | --- | --- |
| № поля | Название поля | Тип поля |
| 1 | id студента | (key)integer |
| 2 | Имя студента | Nvarchar(50) |
| 3 | Фамилия студента | Nvarchar(50) |
| 4 | Группа | Nvarchar(50) |

Таблица 4.1 – T\_ Students

|  |  |  |
| --- | --- | --- |
| № поля | Название поля | Тип поля |
| 1 | id сессии | (key)integer |
| 2 | Дата сессии | smalldatetime |
| 3 | id параметра | integer |
| 4 | id студента | integer |
| 5 | id преподавателя | integer |

Таблица 4.2 – T\_ Labs

|  |  |  |
| --- | --- | --- |
| № поля | Название поля | Тип поля |
| 1 | id преподавателя | (key)integer |
| 2 | Фамилия преподавателя | Nvarchar (50) |
| 3 | Имя преподавателя | Nvarchar (50) |
| 4 | Отчество преподавателя | Nvarchar (50) |

Таблица 4.3 – T\_Prepods

|  |  |  |
| --- | --- | --- |
| № поля | Название поля | Тип поля |
| 1 | id параметра | (key)integer |
| 2 | Значение параметра | Nvarchar(50) |
| 3 | Наименование параметра | Nvarchar(50) |

Таблица 4.4 – T\_ Parameters

В БД должен быть предусмотрен следующий набор скриптов:

* для добавления записей в таблицы БД;
* для удаления записей из таблиц БД;
* для корректировки данных в таблицах БД;

Необходимо так же настроить права для всех групп пользователей. Предусмотреть возможность связи БД с программным интерфейсом. Доступ к изменению БД должен иметь только администратор!

### 4.1.2 Требования к надёжности

Надежность ПП должна обеспечиваться применяемыми методами проектирования, модульной архитектурой построения, высоким уровнем качества программирования, тестированием программного обеспечения.

### 4.1.3 Дополнительные требования

Дополнительные требования следующие:

* среда программирования – MS Visual Studio 2005.NET;
* объектно-ориентированная методология разработки;
* функциональность программного обеспечения;
* надежность работы, защита от некорректно вводимых данных;
* дружественный пользовательский интерфейс, основанный на стандартных элементах управления ОС Windows (обязательно использовать меню, списки, флажки, переключатели, всплывающие подсказки, контекстно-зависимую справку);
* табличная и графическая интерпретация полученных результатов;
* все диаграммы и рисунки в пояснительной записке выполнить в пакете MS Visio 2003;
* нумерацию версий программы проводить в формате 0.XX.

## 4.2 Требования к функциям (задачам), выполняемым ПО

Программное обеспечение должно выполнять набор функций, удовлетворяющих заданным или подразумеваемым потребностям пользователей. Набор указанных функций определяется во внешнем описании ПП.

## 4.3 Требуемые компьютерные версии

В состав ПП и сопроводительных материалов в компьютерной форме, необходимых для сдачи работы, входят:

* функциональная модель (IDEF1X-диаграммы в пакете BPWin 7);
* инфологическая модель (ER-диаграммы в пакете ERWin 7);
* файл MS Office Excel для тестовой загрузки данных в базу;
* пакет Integration Services (IS) для автоматического переноса тестовых данных из MS Office Excel в MS SQL Server 2005;
* пакет Reporting Services (RS) для отображения в численном и графическом виде отчетных показателей на Web-сервере отчетов;
* backup-копия базы данных;
* прикладная программа для сопровождения системы (дистрибутив, исходники);
* внешний help-файл (откомпилированный, исходники).

# 5. ПОРЯДОК КОНТРОЛЯ И ПРИЕМКИ РАБОТЫ

Передача рабочих материалов и документов между Исполнителем и Заказчиком осуществляется по электронной почте. Пересылка по электронной почте должна сопровождаться обязательным подтверждением доставки. Рассмотрение, коррекция и обратная отправка материалов должны выполняться в срок не более 5 (пяти) рабочих дней.

Исполнителем предоставляются следующие компьютерные версии:

* реализация алгоритма расчета в электронных таблицах MS Office Excel 2003;
* файл функциональной модели в пакете BPWin;
* архитектура программного обеспечения, диаграмма классов и эскизы интерфейса в пакете MS Visio 2003;
* файл справки в пакете Help&Manual 5;
* руководство пользователя в форматах \*.hlp, \*.chm, \*.pdf.

Допуск к защите курсовой работы осуществляется только после предварительной сдачи преподавателю всего набора содержимого электронных папок и утвержденной пояснительной записки.

# 6. ТРЕБОВАНИЯ К ПОЯСНИТЕЛЬНОЙ ЗАПИСКЕ

## 6.1 Общие требования к ПЗ

Пояснительная записка (ПЗ) выполняется в соответствии с требованиями по оформлению курсовых и дипломных работ [41]. Состав отдельных разделов регламентируется методическими указаниями [40].

## 6.2 Структура ПЗ

Структура пояснительной записки включает следующее:

* титульный лист;
* бланк задания преподавателя;
* оглавление;
* введение;
* назначение, цели и задачи создания информационной системы;
* архитектура построения информационной системы;
* разработка функциональной модели системы;
* инфологическое и даталогическое моделирование базы данных;
* физическая реализация базы данных, автоматическая загрузка тестовых данных;
* блок-схема алгоритма обработки данных в системе;
* разработка приложения по сопровождению базы данных;
* разработка системы отображения отчетов;
* функциональные возможности информационной системы;
* заключение;
* список литературы;
* приложение (техническое задание на проект, подписанное преподавателем; руководство администратора; руководство пользователя информационной системы).

**6.3 Особые дополнительные требования**

* **Функциональная модель:** выполняется по стандарту IDEF1X в пакете BPWin 7.
* **Концептуальная модель:** выполняется в виде ER-диаграмм в пакете ERWin 7.
* **Даталогическая модель:** генерируется из CASE-программы в целевую СУБД.
* **Пакет Integration Services (IS):** пакет автоматической передачи тестовых данных из MS Office Excel в базу данных MS SQL Server 2005 должен быть реализован с использованием службы интеграции (Integration Services) сервера MS SQL Server 2005. Пакет IS разработать в среде Business Intelligence Development Studio, входящей в состав MS SQL Server 2005.
* **Пакет Reporting Services (RS):** создание пакетов Reporting Services для отображения в численном и графическом виде отчетных показателей выполнить в среде Business Intelligence Development Studio, входящей в состав MS SQL Server 2005. Численное представление отчетных данных с использованием инструмента Reporting Services должно включать в себя использование сводных таблиц, интерактивных элементов (выпадающих списков, календарей, расширяемых областей отчета и пр.) для диалога с пользователем. Графическое представление отчетных данных с использованием инструмента Reporting Services должно включать в себя использование трендовых диаграмм (Chart), интерактивных элементов (выпадающих списков, календарей и пр.) для диалога с пользователем. На диаграмме предусмотреть возможность отображения нескольких линий трендов для сравнения динамики изменения отчетных показателей.
* **Обработка данных:** выполняется средствами прикладной программы и СУБД. В случае использования MS SQL Server 2005 обработку выполнить преимущественно с помощью хранимых процедур (stored procedures) и организации представлений (views). Взаимодействие прикладной программы и сервера организовать с помощью SQL-запросов и вызова хранимых процедур.
* **Контроль доступа:** организация работы нескольких пользователей с различными привилегиями (users). Предусмотреть минимум две категории пользователей:

1) пользователь для сопровождения всех таблиц базы данных, включая блок НСИ;

2) пользователь для реализации основных функций бизнес-логики.

* **Прикладная программа:** разработка в среде программирования Visual Studio 2005 .NET; комментарии в исходном коде; контроль данных при заполнении форм; табличная и графическая интерпретации полученных результатов; контекстно-зависимая справка.

### 7. ИСТОЧНИКИ РАЗРАБОТКИ

1. Балена Ф. Современная практика программирования на Microsoft Visual Basic и Visual C#: пер. с англ. / Ф. Балена, Дж. Димауро. М.: Русская редакция, 2006. 640 с.
2. Благовещенская М.М. Информационные технологии систем управления технологическими процессами / М.М.Благовещенская, Л.А.Злобин. М.: Высшая школа, 2005. 768 с.
3. Боггс Уэнди. UML и Rational Rose 2002 / Уэнди Боггс, Майкл Боггс М: ЛОРИ, 2004. – 528 с.
4. Боэм В. Характеристика качества программного обеспечения / В.Боэм, Д.Браун. М.: Мир, 2001.
5. Бутаков Е.А. Методы создания качественного программного обеспечения ЭВМ – М.: Энергоатомиздат, 1999.
6. Буч Г. Объектно-ориентированное проектирование с примерами применения / Г.Буч. М.: Бином, 1998. 550с.
7. Буч Г. Язык UML. Руководство пользователя / Г.Буч, Д.Рамбо, А.Джекобсон / М.: ДМК, 2000. 432 с. (есть электронный вариант).
8. Ван-Тассел Д. Стиль, разработка, эффективность, отладка и испытание программ – М.: Мир, 2001.
9. Вельбицкий И.В. Технология программирования / И.В.Вельбицкий. Киев, 1998.
10. Вендров А.М. Проектирование программного обеспечения экономических информационных систем: Учебник / А.М.Вендров. М.: Финансы и статистика, 2002. 352 с.
11. Вирт Н.П. Алгоритмы и структуры данных / Н.П.Вирт. M: Мир, 1998. 320 с.
12. Гласс Р. Руководство по надежному программированию / Р.Гласс. М.: Финансы и статистика, 2002.
13. ГОСТ 24104-85 ЕСС АСУ «Автоматизированные системы управления. Общие требования»;
14. ГОСТ 24601-86 ЕСС АСУ «Автоматизированные системы управления. Стадии создания».
15. ГОСТ 34.201-89 «Информационная технология. Комплекс стандартов на автоматизированные системы. Виды, комплектность и обозначение документов при создании автоматизированных систем»;
16. ГОСТ 34.602-89 «Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы»;
17. ГОСТ 34.603-92 «Информационная технология. Комплекс стандартов на автоматизированные системы. Виды и порядок освоения»;
18. Дал У. Структурное программирование / У.Дал, Э.Дейкстра, К.Хоор. М.: Мир, 2000.
19. Дейкстра Э.В. Заметки по структурному программированию / У. Дал, Э. Дейкстра, К.Хоор. В составе сборника «Структурное программирование». М.: Мир, 1975. С. 7–97. <http://khpi-iip.mipk.kharkiv.edu/library/extent/dijkstra/ewd249/index.html> .
20. Зелковиц М. Принципы разработки ПО / М.Зелковиц, А.Шоу. – М.: Мир, 2002.
21. Калянов Г.Н. CASE-технологии и консалтинг в автоматизации бизнес-процессов / Г.Н.Калянов. М.: Горячая линия – Телеком, 2000. – 320 с.
22. Карпов Б. Visual Basic 6: Специальный справочник / Б.Карпов. СПб.: Питер, 2002. 416 с.
23. Кауфман В.Ш. Языки программирования. Концепции и принципы / В.Ш.Кауфман. М.: Радио и связь, 1993.
24. Кватрани Т. Визуальное моделирование с помощью Rational Rose 2002 и UML / Т.Кватрани. М.: Издательский дом «Вильямс», 2003. 192 с.
25. Керниган Б. Практика программирования / Б.Керниган, Р.Пайк. СПб.: Невский диалект, 2001.
26. Кнут Д. Искусство программирования. Т.1. Основные алгоритмы (3-е изд.) // М.: Издательский дом «Вильямс», 2000. 720 с.
27. Коржинский С. Работа на компьютере. Популярный самоучитель / С.Коржинский. СПб.: Питер, 2005. 368 с.
28. Ларман К. Применение UML 2.0 и шаблонов проектирования [Текст]: [пер. с англ.] / К.Ларман. – М.: Вильямс, 2007. – 736 с.
29. Липаев В.В. Качество программного обеспечения / В.В. Липаев. М.: Финансы и статистика, 2003.
30. Майерс Г. Надежность программного обеспечения. [Текст] : [пер. с англ.] / Под ред. В.Ш.Кауфмана // М.: Мир, 1980. 360 с.
31. Макконнелл C. Совершенный код. Мастер-класс [Текст]: [пер. с англ.] / С.Макконнелл. – СПб.: Питер, 2007. – 896 с.
32. Марка Д.А. Методология структурного анализа и проектирования SADT / Д.А.Марка, К.МакГауэл. М.: Метатехнология, 1993. 346 с.
33. Мейер Б. Методы программирования. В 2-х т. / Б.Мейер, К.М.Бодуэн. М.: Мир, 1997.
34. Петров В.Н. Информационные системы / В.Н.Петров. СПб.: Питер, 2003. 688 с.
35. Рейнвотер Дж. Как пасти котов. Наставление для программистов, руководящих другими программистами [Текст]: [пер. с англ.] / Дж.Рейнвотер. – СПб.: Питер, 2007. – 256 с.
36. Фаулер М. Архитектура корпоративных программных приложений [Текст]: [пер. с англ.] / М.Фаулер. – М.: Вильямс, 2006. – 544 с.
37. Фридман А.Л. Основы объектно-ориентированной разработки программ / А.Л.Фридман. М.: Финансы и статистика, 2000. 192 с.
38. Чарнецки К. Порождающее программирование: методы, инструменты, применение. Для профессионалов. [Текст] : [пер. с англ.] / К.Чарнецки, У.Айзенекер. – СПб.: Питер, 2005. – 731 с.
39. Лавров В.В. Технология разработки программного обеспечения: методические указания к выполнению курсовой работы для студентов обучающихся по направлению 230200 – «Информационные системы» специальности 230201 – «Информационные системы и технологии»/ В.В.Лавров, Кисилев Е.В. – Екатеринбург: УГТУ–УПИ, 2008. – 59 с.
40. Лошкарев Н.Б. Указания к оформлению дипломных и курсовых проектов и работ. Методические указания [Текст] / Н.Б.Лошкарев, А.Н.Лошкарев, Л.А.Зайнуллин. – Екатеринбург: ГОУ ВПО УГТУ – УПИ, 2007. – 49 с.

ТЗ СОСТАВИЛ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Наименование  организации  (предприятия) | Должность  Исполнителя | Фамилия, имя,  отчество (полностью) | Подпись | Дата |
| ФГАОУ ВПО «УрФУ им Б.Н.Ельцина» | Студент, Мт-46052 | Бородулин Юрий Александрович |  |  |

ТЗ СОГЛАСОВАНО

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Наименование  организации  (предприятия) | Должность  Заказчика | Фамилия, имя,  отчество (полностью) | Подпись | Дата |
| ФГАОУ ВПО «УрФУ им Б.Н.Ельцина» | доцент кафедры ТИМ, к.т.н. | Лавров Владислав Васильевич |  |  |

# ПРИЛОЖЕНИЕ Б «Фрагменты листинга формы frmMain.cs»

|  |
| --- |
| public frmMain()  {  frm\_Reg DataBaseAcces = new frm\_Reg();  DialogResult diagResult = DataBaseAcces.ShowDialog();  if (diagResult == DialogResult.OK)  {  bStart = true;  cs = DataBaseAcces.ConnectionString;  UserID = DataBaseAcces.UserID;  InitializeComponent();  CenterToScreen();  }  else if (diagResult == DialogResult.Cancel)  {  this.Dispose();  }  else  {  System.Diagnostics.Debug.Print("Неизвестный результат работы диалогового окна");  this.Dispose();  }  }  /// <summary>  /// Метод первоначальной загрузки формы  /// </summary>  private void frmMain\_Load(object sender, EventArgs e)  {  // TODO: This line of code loads data into the 'dBlabDataSet.T\_Labs' table. You can move, or remove it, as needed.  //this.t\_LabsTableAdapter.Fill(this.dBlabDataSet.T\_Labs);  // TODO: This line of code loads data into the 'dBlabDataSet.T\_Labs' table. You can move, or remove it, as needed.  //this.t\_LabsTableAdapter.Fill(this.dBlabDataSet.T\_Labs);  FormOptionDefault();  // Загрузить данные из источника  ChooseDataStorage(\_bDataFromDB);  //this.reportViewer1.RefreshReport();  //this.reportViewer1.RefreshReport();  }  /// <summary>  /// Метод первоначальной настройки элементов управления формы  /// </summary>  private void FormOptionDefault()  {  tc.TabPages.Clear();  tc.TabPages.Add(tabPage1);    // Создать соединение с базой данных DBlab.  SqlConnection cn = new SqlConnection();  cn.ConnectionString = cs.ToString();  Program.DBlabConnectionString = cs.ToString();  try  {  dBlabDataSet.Clear();  // TODO: This line of code loads data into the 'dBlabDataSet.T\_Parameters' table. Youcan move, or remove it, as needed.  this.t\_ParametersTableAdapter.Fill(this.dBlabDataSet.T\_Parameters);  // TODO: This line of code loads data into the 'dBlabDataSet.T\_Stusents' table. You can move, or remove it, as needed.  this.t\_StudentsTableAdapter.Fill(this.dBlabDataSet.T\_Student);  // TODO: This line of code loads data into the 'dBflabDataSet.T\_Prepods' table. You can move, or remove it, as needed.  this.t\_ PrepodsTableAdapter.Fill(this.dBlabDataSet.T\_Prepods);  // TODO: This line of code loads data into the 'dBlabDataSet.T\_Labs' table. You can move, or remove it, as needed.  this.t\_LabsTableAdapter.Fill(this.dBlabDataSet.T\_Labs);  }  catch (Exception ex)  {  ReportErrorMessage(ex);  }  finally |

|  |
| --- |
| {  cn.Close(); // В любом случае закрыть соединение с базой данных  }  }  /// <summary>  /// Метод выбора источника загрузки данных в приложение  /// </summary>  /// <param name="bDataFromDB">Загрузить данные из базы данных (true) или из кода для отладки (false)</param>  private void ChooseDataStorage(bool bDataFromDB)  {  if (bDataFromDB)  {  ImportData();  }  else  {  //FillDataTest();  }  }  /// <summary>  /// Метод выводит сообщение об ошибке на основании кода ошибки, полученного от вызывающей процедуры.  /// Способ вывода - на экран и в журнал.  /// </summary>  /// <param name="errorToReport">Описание ошибки</param>  private void ReportErrorMessage(Exception errorToReport)  {  // Показать на экране сообщение, соответствующее коду ошибки.  MessageBox.Show(errorToReport.Message);  // Записать факт ошибки в журнал  // Открыть (или создать) файл для вывода информации.  // Файл для чтения и записи, но без возможности совместного использования.  // Сохранить указатель на файл в объекте FileStream  FileInfo f2 = new FileInfo(strErrorLog);  StreamWriter writer = f2.AppendText();  writer.Write(writer.NewLine);  writer.WriteLine("---- Дата: " + DateTime.Now.Day.ToString() + "-" +  DateTime.Now.Month.ToString() + "-" + DateTime.Now.Year.ToString() + " " +  DateTime.Now.Hour.ToString() + ":" + DateTime.Now.Minute.ToString());  writer.WriteLine(errorToReport.Message);  writer.Close(); // Очистить буфер!  }  /// <summary>  /// Метод загрузки данных из БД и отображение их на гридах  /// </summary>  private void ImportData()  {  FillData(); // Загрузить данные из БД в DataSet.  SetDataViews(); // Настроить представления.  SetDataSourceDGW(); // Перерисовать гриды.  }  /// <summary>  /// Метод заполнения таблиц данными  /// </summary> |