

Федеральное государственное бюджетное образовательное
учреждение
высшего профессионального образования
Национальный исследовательский университет «МЭИ»

Отчет
к курсовому проекту
по дисциплине «Базы данных»
Тема: Разработка БД для организации преддипломной практики на
кафедре

Студент:

Филимонова А.А.

фамилия, и., о.

А-13-00

группа

подпись

Преподаватель: доц., к.т.н. Сидорова Н.П.

должность, звание, фамилия, и., о.

подпись

Москва 2012

Задание на курсовой проект

Цель работы: разработка БД для поддержки процесса организация преддипломной практики студентов кафедры.

БД должна поддерживать выполнение следующих функций:

- Формирование списка баз практики;
- Назначение руководителя практики от кафедры;
- Назначение руководителя практики от предприятия;
- Распределение студентов по базам практики;
- Организация контроля прохождения практики руководителем от кафедры;
- Контроль защиты отчета по практике на кафедре.

Исходные данные для выполнения курсового проекта: среда разработки – Microsoft Windows NT; AllFusion Modeler Suite, СУБД по выбору студента, язык программирования SQL, язык высокого уровня по выбору студента.

Литература

1. Пирогов В.Ю. SQL Server 2005: программирование клиент-серверных приложений. – СПб: БХВ-Петербург, 2006. – 336 с.
2. Н.П. Сидорова. Базы данных. Методические указания к курсовому проектированию. – М.: Изд. дом МЭИ, 2010. – 40 с.

Оглавление

Задание на курсовой проект	2
Сокращения.....	4
Введение.....	5
1. Анализ предметной области	6
1.1. Описание предметной области.....	6
1.2. Анализ функций процесса	6
2. Проектирование базы данных	9
2.1. Разработка ER-модели	9
2.2. Анализ реляционной модели.....	9
3. Реализация базы данных	13
4. Программная реализация	17
4.1. Объектная модель ADO .NET	17
4.2. Примеры экранных форм интерфейса.....	18
Выводы по работе	19
Литература	20
Приложение 1	21
Приложение 2	23

Сокращения

БД – база данных

3НФ – третья нормальная форма

СУБД – система управления базами данных

ФЗ – функциональная зависимость

Введение

Целью курсового проекта является разработка моделей базы данных (БД) и интерфейсных средств для выполнения указанных в задании функций. БД обеспечивает хранение информации и представляет собой поименованную совокупность данных, организованных по определенным правилам, включающим общие принципы описания, хранения и манипулирования данными. БД является информационной моделью предметной области, в которой решаются поставленные задачи.

Этапы выполнения курсового проекта:

- поведение анализа предметной области;
- проектирование БД;
- реализация БД средствами выбранной СУБД;
- программная реализация интерфейса с БД.

Для анализа предметной области используется CASE-средство AllFusion Process Modeler, который входит в состав пакета AllFusion Modeler Suite. Проектирование моделей БД выполняется с помощью CASE-средства AllFusion ERwin Data Modeler, входящий в состав этого же пакета.

Реализация БД проводится в среде Microsoft SQL Server. В качестве языка реализации был выбран C# в среде Microsoft Visual Studio.

1. Анализ предметной области

1.1. Описание предметной области

Предметная область курсового проекта: распределение студентов на преддипломную практику на кафедре. Студенты могут проходить преддипломную практику в организации или на кафедре. Для каждого студента на кафедре назначается руководитель, который осуществляет контроль прохождения практики и принимает отчет по её результатам.

Если студент проходит практику на предприятии, то назначается руководитель практики на нем. В этом случае руководитель практики от кафедры контролирует график выполнения практики и результаты его выполнения.

При организации процесса прохождения преддипломной практики определены следующие правила:

- Студент может проходить практику только на одной базе практики.
- У одного студента может быть только один руководитель практики от кафедры.
- У одного студента может быть только один руководитель практики от предприятия.
- Один руководитель практики от кафедры может руководить несколькими студентами на одной базе практики.
- Руководителя практики от предприятия может руководить несколькими студентами.
- Результатом прохождения практики является оценка: удовл., хорошо, отлично.

1.2. Анализ функций процесса

Для проведения анализа предметной области используется методология IDEF. В рамках этой методологии разрабатывается совокупность диаграмм процессов, которые должны поддерживаться БД. Для построения моделей использовалось CASE-средство AllFusion Process Modeler. Первой разрабатывается контекстная диаграмма (рис.1.1), на которой определены входные и выходные данные процесса.

Распределение студентов на преддипломную практику предполагает выполнение функций, перечисленных в задании. Для каждой функции в ходе выполнения курсового проекта были разработаны модели. На рис. 1.2. приведена модель функции «Формирование баз практики. Модели остальных функций процесса приведены в Приложении 1.

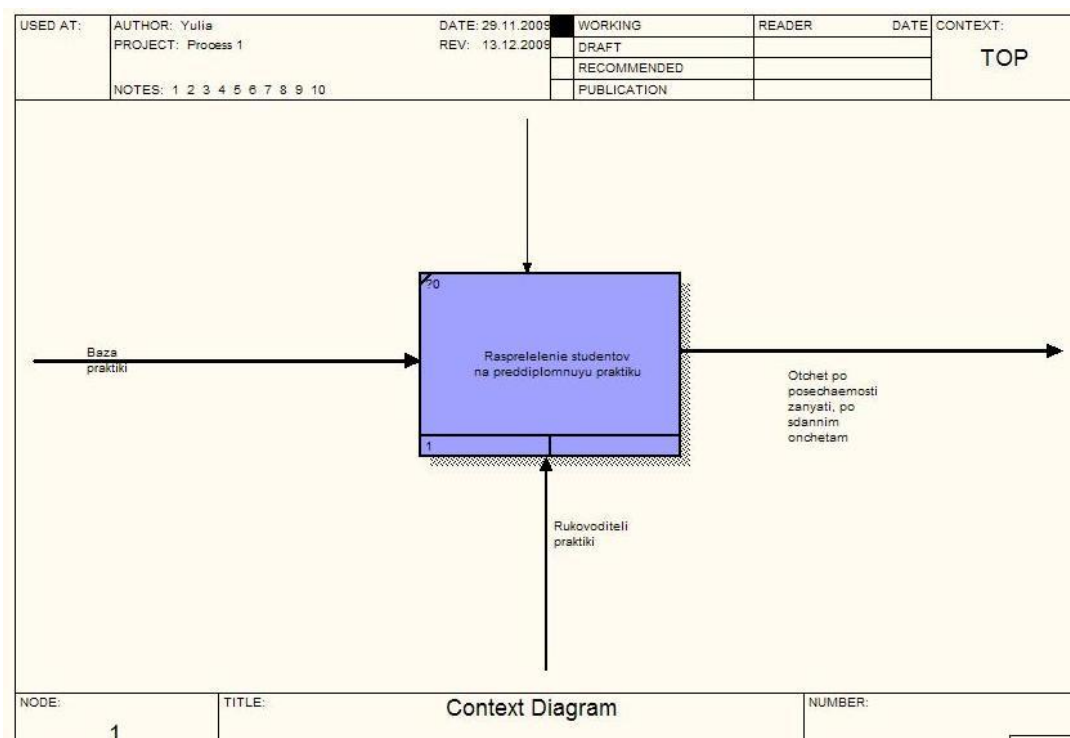


Рис. 1.1. Контекстная диаграмма процесса «Организация преддипломной практики»

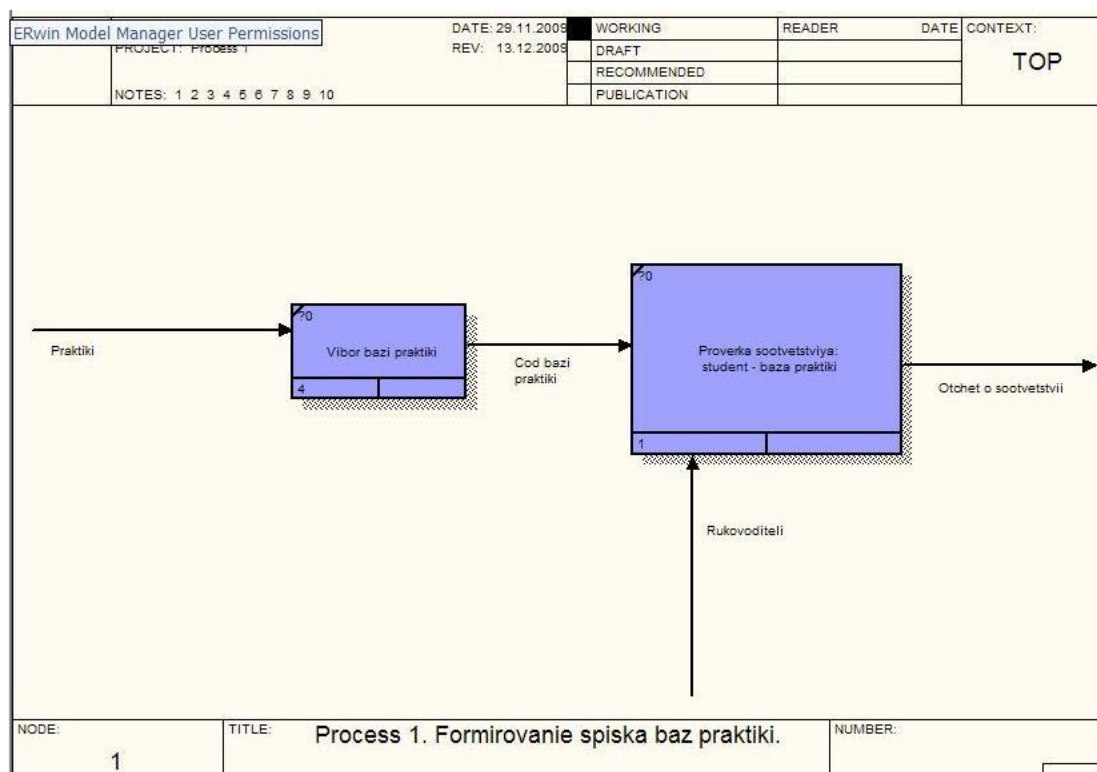


Рис.1.2. Модель функции «Формирование списка баз практики».

Анализ построенных моделей позволил выделить информационные объекты предметной области. Состав и характеристика информационных объектов приведена в табл. 1.

Таблица 1. Информационные объекты и их свойства

№	Имя объекта	Описание объекта	Свойства объекта
1.	Студент	Содержит описание характеристик студента	Номер студента Имя студента Номер группы Контактный телефон
2.	База практики	Содержит характеристики предприятия, на котором проходит практика	Название организации Адрес Контактный телефон
3.	Руководитель1	Руководитель практики от кафедры	Фамилия имя отчество Должность Контактный телефон
4.	Руководитель2	Руководитель практики от предприятия	Фамилия имя отчество Номер отдела Название организации Контактный телефон
5.	Защита	Данные о прохождении защиты отчета	Дата Оценка Аудитория
	Договор	Договор между кафедрой и организацией	Номер договора Дата подписания Ответственный исполнитель Срок действия Контактный телефон

2. Проектирование базы данных

Проектирование БД предполагает разработку моделей нескольких типов:

- Модель «сущность – связь», которая строится на основе выделенных информационных объектов и определения связей между ними.
- Реляционная модель, которая формируется на основе модели «сущность – связь».

2.1. Разработка ER-модели

На основе выделенных информационных объектов средствами AllFusion ERwin Data Modeler была построена модель «сущность – связь» (рис. 2.1.).

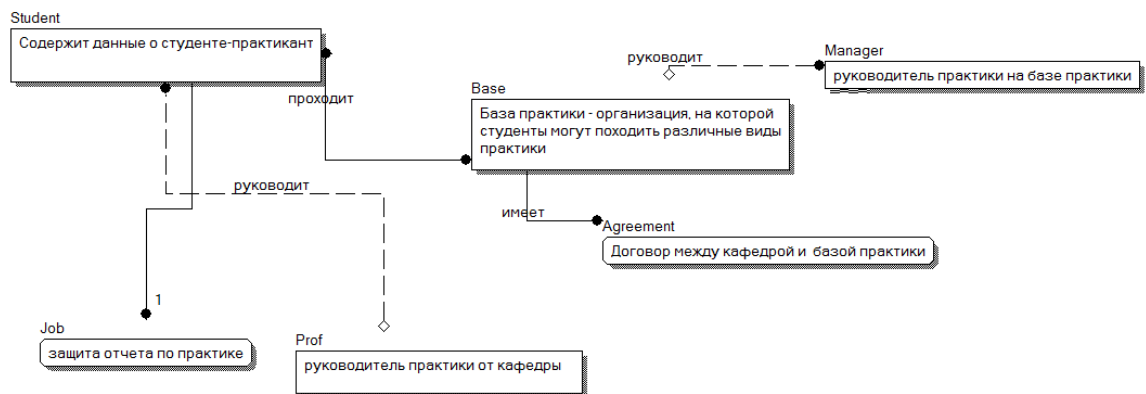


Рис. 2.1. Модель «сущность-связь» для БД

Документирование модели было выполнено с помощью CASE-средства Erwin Data Modeler и приведено на CD-диске в файле ermod.pdf .

2.2. Анализ реляционной модели

На основе разработанной ER-модели была построена реляционная модель, представленная на рис. 2.2.

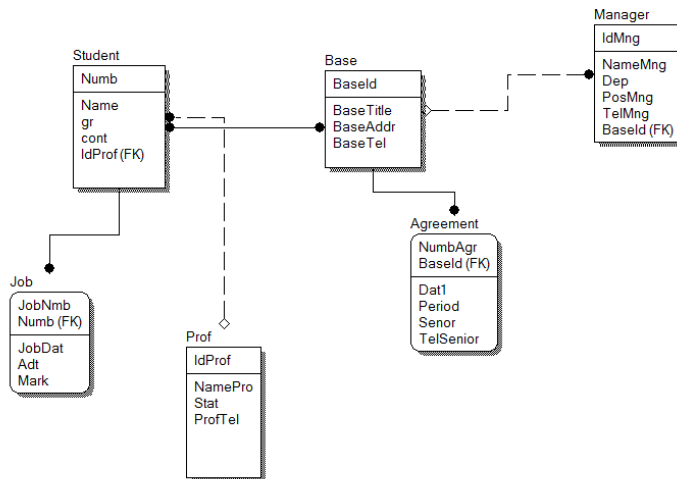


Рис. 2.2 Реляционная модель БД

Для получения хороших характеристик работы с БД, она должна находиться в 3НФ. Для проверки этого необходимо обеспечить атомарность атрибутов, выделить функциональные зависимости (ФЗ) между атрибутами в каждом отношении и провести их анализ.

Отношение **Student**(Numb,Name,gr,cont,IdProf), ключ отношения – Numb.

Функциональные зависимости:

Numb→Name

Numb→gr

Numb→cont

Numb→IdProf

Отношение **Prof**(IdProf, NameProf, Stat, Proftel), ключ – IdProf.

Функциональные зависимости:

IdProf → NameProf,

IdProf →Stat,

IdProf →Proftel

Отношение **Agreement**(NumdAgr, BaseId, Dat1, Period,Senor,TelSenor), ключ - NumdAgr

Функциональные зависимости:

NumdAgr → BaseId,

NumdAgr → Dat1,

NumdAgr → Period,

NumdAgr → Senior,

NumdAgr → TelSenior

Отношение **Base**(BaseId, BaseTitle, BaseAddr), ключ – BaseId

Функциональные зависимости:

BaseId → BaseTitle,

BaseId → BaseAddr

Отношение **Manager**(IdMng, NameMng, Dep, PosMng, TelMng, BaseId),
ключ – IdMng

Функциональные зависимости:

IdMng → NameMng,

IdMng → Dep,

IdMng → PosMng,

IdMng → TelMng,

IdMng → BaseId

Отношение **Job**(JobNmb, Numb, JobDat, Adt, Mark), ключ - JobNmb, Numb

Функциональные зависимости:

JobNmb, Numb → JobDat,

JobNmb, Numb → Adt,

JobNmb, Numb → Mark

Других ФЗ для отношений не выявлено. Обоснование нахождения БД в ЗНФ предполагает проверку ограничений, накладываемых в ЗНФ.

1. Все таблицы БД находятся в 1НФ, их поля содержат атомарные значения.

2. Полная ФЗ неключевых атрибутов от ключа для отношений **Student, Prof, Agreement, Base, Manager** обеспечивается тем, что ключи в этих отношениях простые.

3. Для отношения **Job** полная ФЗ между неключевыми атрибутами и составным ключом существует.

4. В разработанных отношениях отсутствуют транзитивные зависимости.

Следовательно, БД находится в ЗНФ.

Полное описание разработанной реляционной модели создано с помощью CASE-средства Erwin Data Modeler и приведено на CD-диске в файле rmdmod.pdf.

3. Реализация базы данных

В качестве СУБД для реализации БД в курсовом проекте была выбрана СУБД Microsoft SQL Server. Эта СУБД разработана корпорацией Microsoft и поддерживает реляционную модель БД. Основным используемый язык запросов – Transact-SQL, созданный совместно Microsoft и Sybase. Transact-SQL является реализацией стандарта ANSI/ISO структурированного языка запросов (SQL) с расширениями. СУБД SQL Server используется для работы со средними и большими БД. Ниже приведено описание БД средствами Transact-SQL.

```
CREATE RULE Validation_Rule_429
    AS @col IN ('удовл', 'хорошо', 'отлично')
go
CREATE RULE ch2
    AS @col IN ('acc', 'ст.пр', 'преп', 'доц', 'проф')
go
CREATE TABLE Agreement
(
    NumbAgr      integer NOT NULL ,
    Dat1         datetime NULL ,
    Period       integer NULL ,
    Senor        varchar(20) NULL ,
    TelSenior    char(18) NULL ,
    BaseId       integer NOT NULL
)
go
ALTER TABLE Agreement
    ADD CONSTRAINT XPKAgreement PRIMARY KEY CLUSTERED
(NumbAgr ASC,BaseId ASC)
go
CREATE TABLE Base
(
    BaseId       integer NOT NULL ,
    BaseTitle    varchar(20) NULL ,
    BaseAddr     varchar(20) NULL ,
    BaseTel      char(18) NULL
```

```

)
go
ALTER TABLE Base
    ADD CONSTRAINT XPKBase PRIMARY KEY CLUSTERED (BaseId ASC)
go
CREATE TABLE Job
(
    JobNmb        integer NOT NULL ,
    JobDat        datetime NULL ,
    Adt           varchar(20) NULL ,
    Numb          varchar(20) NOT NULL ,
    Mark          integer NULL
)
go
ALTER TABLE Job
    ADD CONSTRAINT XPKJob PRIMARY KEY CLUSTERED (JobNmb
ASC,Numb ASC)
go
CREATE TABLE Manager
(
    IdMng         integer NOT NULL ,
    NameMng       varchar(20) NULL ,
    Dep           integer NULL ,
    PosMng        integer NULL ,
    TelMng        varchar(20) NULL ,
    BaseId        integer NULL
)
go
ALTER TABLE Managers
    ADD CONSTRAINT XPKManager PRIMARY KEY CLUSTERED (IdMng
ASC)
go
CREATE TABLE Prof
(
    IdProf        integer NOT NULL ,
    NameProf      varchar(20) NULL ,

```

```

        Stat          integer NULL ,
        ProfTel       varchar(20) NULL
    )
go
ALTER TABLE Prof
    ADD CONSTRAINT XPKProf PRIMARY KEY CLUSTERED (IdProf ASC)
go
CREATE TABLE Student
(
    Name          varchar(20) NULL ,
    gr            varchar(20) NULL ,
    cont          varchar(20) NULL ,
    Numb          varchar(20) NOT NULL ,
    IdBase        integer NULL,
    IdProf        integer NULL
)
go
ALTER TABLE Student
    ADD CONSTRAINT XPKStudent PRIMARY KEY CLUSTERED (Numb ASC)
go
ALTER TABLE Agreement
    ADD CONSTRAINT содержит FOREIGN KEY (BaseId) REFERENCES
Base(BaseId)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION
go
ALTER TABLE Job
    ADD CONSTRAINT осуществляет FOREIGN KEY (Numb) REFERENCES
Student(Numb)
        ON DELETE NO ACTION
        ON UPDATE NO ACTION
go
ALTER TABLE Manager
    ADD CONSTRAINT руководит FOREIGN KEY (BaseId) REFERENCES
Base(BaseId)
        ON DELETE NO ACTION

```

```

        ON UPDATE NO ACTION

go
ALTER TABLE Student
    ADD CONSTRAINT руководит2 FOREIGN KEY (IdProf) REFERENCES
Prof(IdProf)

        ON DELETE NO ACTION
        ON UPDATE NO ACTION

go
exec sp_bindrule 'Validation_Rule_429', 'Job.Mark'

go
exec sp_bindrule 'ch2', 'Prof.IdProf'

go
exec sp_bindrule 'ch2', 'Prof.Stat'

go

```


4. Программная реализация

Для реализации приложения с БД выбран язык C# и технология ADO.NET.

4.1. Объектная модель ADO .NET

ADO .NET (ActiveX Data Objects .NET) является набором классов, реализующих программные интерфейсы для облегчения подключения к БД из приложения, независимо от особенностей реализации конкретной СУБД. ADO .NET широко используется совместно с технологией web-программирования с использованием объектов ASP .NET для доступа к расположенным на сервере базам данных со стороны клиента.

Объектная модель ADO .NET реализует отсоединенный доступ к данным. При этом в Visual Studio .NET существует множество встроенных мастеров и дизайнеров, которые позволяют реализовать механизмы доступа к БД еще на этапе разработки программного кода. С другой стороны, задача получения доступа к данным может быть решена непосредственно во время выполнения приложения.

Концепция доступа к данным в ADO .NET основана на использовании двух компонентов:

- набора данных (представляется объектом класса DataSet) со стороны клиента. Это локальное временное хранилище данных;
- провайдера данных (представляется объектом класса DataProvider). Это посредник, обеспечивающий взаимодействие приложения и базы данных со стороны базы данных (в распределенных приложениях – со стороны сервера).

Объектная модель ADO .NET предполагает существование (при написании приложения для работы с базой данных – использование) двух множеств классов, выполняющих четко определенные задачи при работе с базой данных (рис. 4.1.)

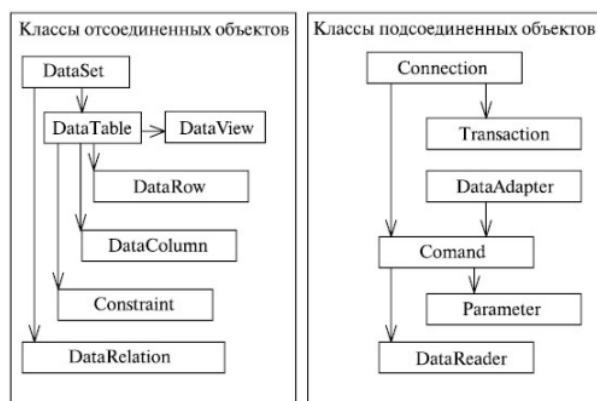


Рис. 4.1. Состав классов технологии ADO.NET

4.2. Примеры экранных форм интерфейса

Основная форма разработанного приложения представлена на рис. 4.2.

firstName	lastName	patronymic	practiceMark	protectionMark	baseName
Алексей	Медвенцев	Владимирович	4	4	ОАО "РЖД"
Надежда	Недоукина	Григорьевна	5	5	ОАО "РЖД"
Милана	Гурьянова	Николаевна	5	4	ОАО "РЖД"
Павел	Рогозин	Константинович	4	4	ОАО «Газпром»
Светлана	Гандыбина	Власиевна	3	3	ОАО «Газпром»
Антон	Молостнов	Кириллович	4	5	ОАО "АВТОВАЗ"
Елизавета	Недачина	Витальевна	3	4	ОАО "АВТОВАЗ"

Рис. 4.2. Главная форма программного приложения

При нажатии на кнопку «Add practice» можно добавить новую базу практики. При этом открывается диалоговое окно, со списком студентов, не привязанных ни к одной из практик, а также с полями для ввода названия и адреса новой базы практики (рис. 4.3). Добавление новой базы практики выполняется с помощью запроса:

```
@ "insert into Base(baseTitle,baseAddr,managerId,profId)values ('" +  
baseName_tb.Text + "',''" + address_tb.Text + "',''" + managerId + "','" + professorId + "')
```

Полностью файлы проекта приведены на CD-диске в файле StudingPractice. Разработанная программа реализует следующие функции с БД:

- Добавление новой базы практики в таблицу Base;
- Добавление данных о руководителе от кафедры в таблицу Prof;
- Добавление данных о руководителе практики от предприятия в таблице Manager;
- Изменение данных о базе практики (IdBase) в таблице Student;
- Составление списка студентов, не распределенных на практику по прохождению практики;
- Добавление данных об оценках по практике в таблицу Job;
- Поиск договора с базой практики.

Примеры экранных форм проекта приведены в Приложении 2

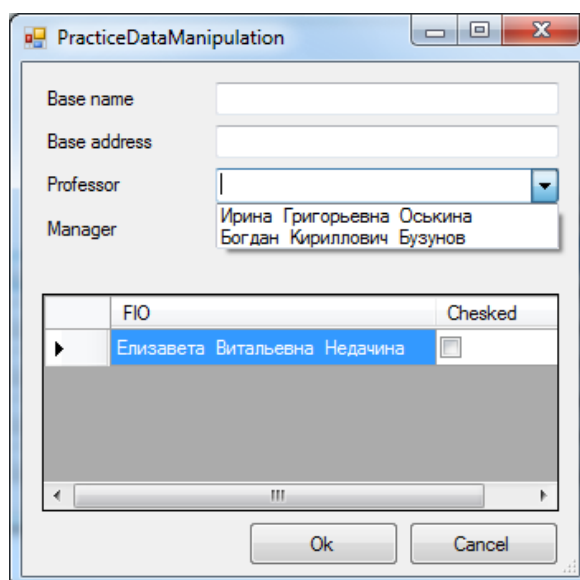


Рис. 4.3. Добавление Базы практики для студента

Выводы по работе

В результате выполнения курсового проекта:

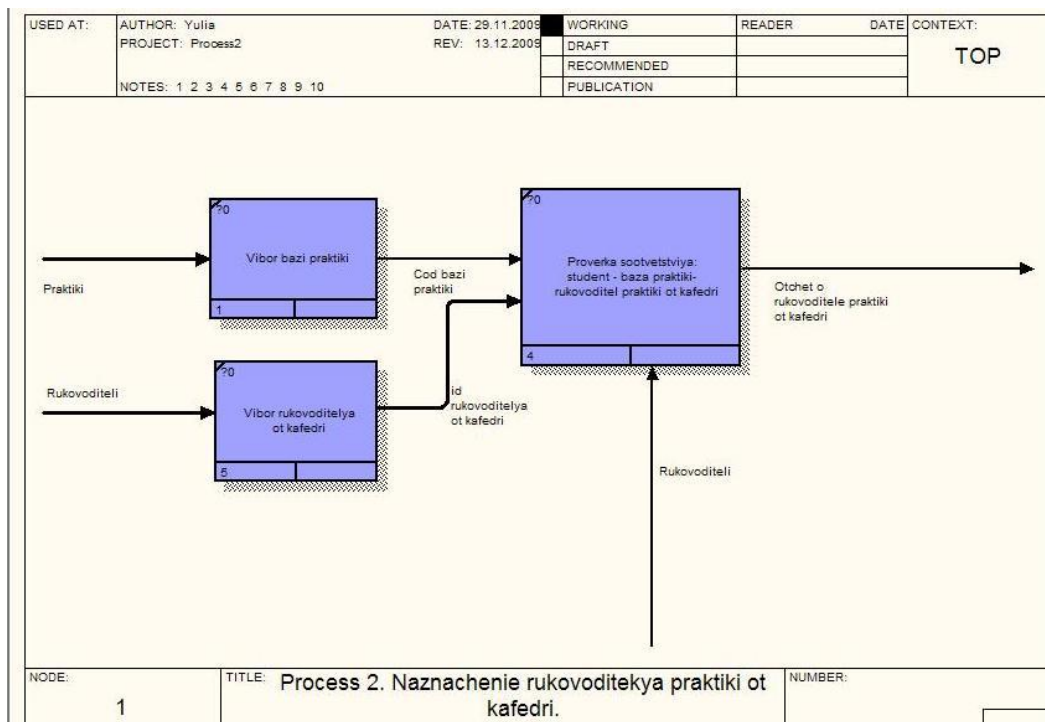
1. Проведен анализ процесса организации преддипломной практики студентов
2. Спроектирована БД из шести таблиц. Модель БД доведена до требуемой ЗНФ.
3. Разработаны программные средства интерфейса с БД и приложения, реализующие указанные в задании функции. Объем программного кода составляет 300 операторов языка C#.

Литература

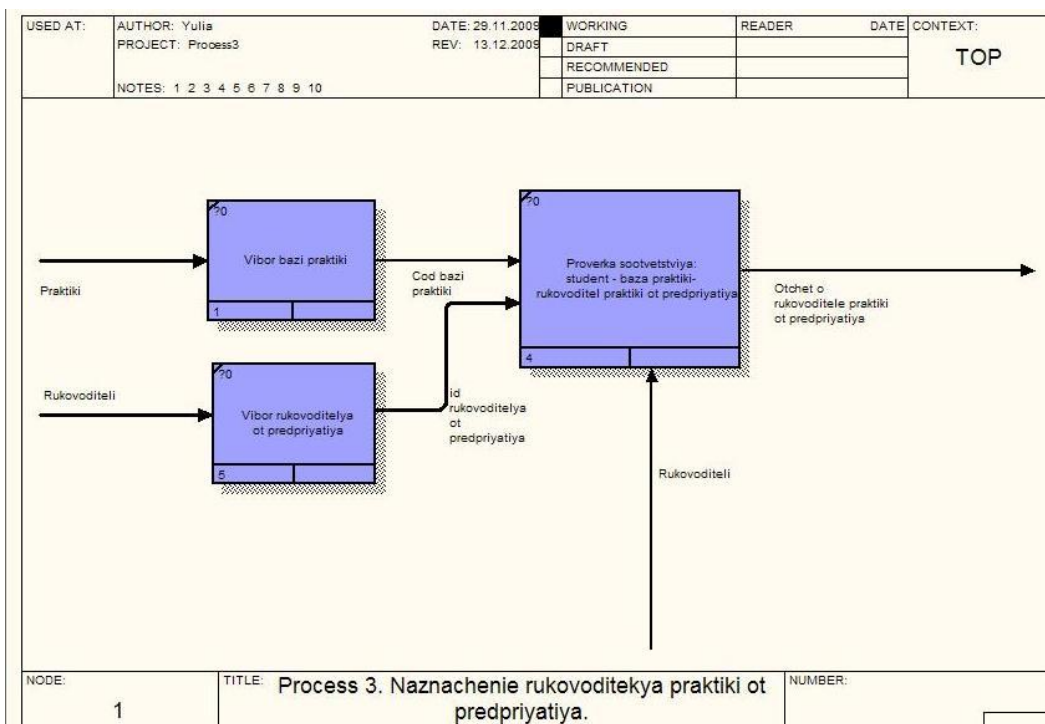
1. Пирогов В.Ю. SQL Server 2005: программирование клиент-серверных приложений. – СПб: БХВ-Петербург, 2006. – 336 с.
2. Н.П. Сидорова. Базы данных. Методическое пособие к курсовому проектированию. – М.: Изд-во МЭИ, 2010. – 40 с.
3. Маран М.М. Программирование на языке C# в среде Microsoft Visual Studio 2005: Учебное пособие по курсу. – М. : Изд-во МЭИ, 2007. – 64 с.
4. Пауэрс Л. Microsoft Visual Studio 2008. – СПб: БХВ-СПб, 2009. – 1200 с.

Приложение 1

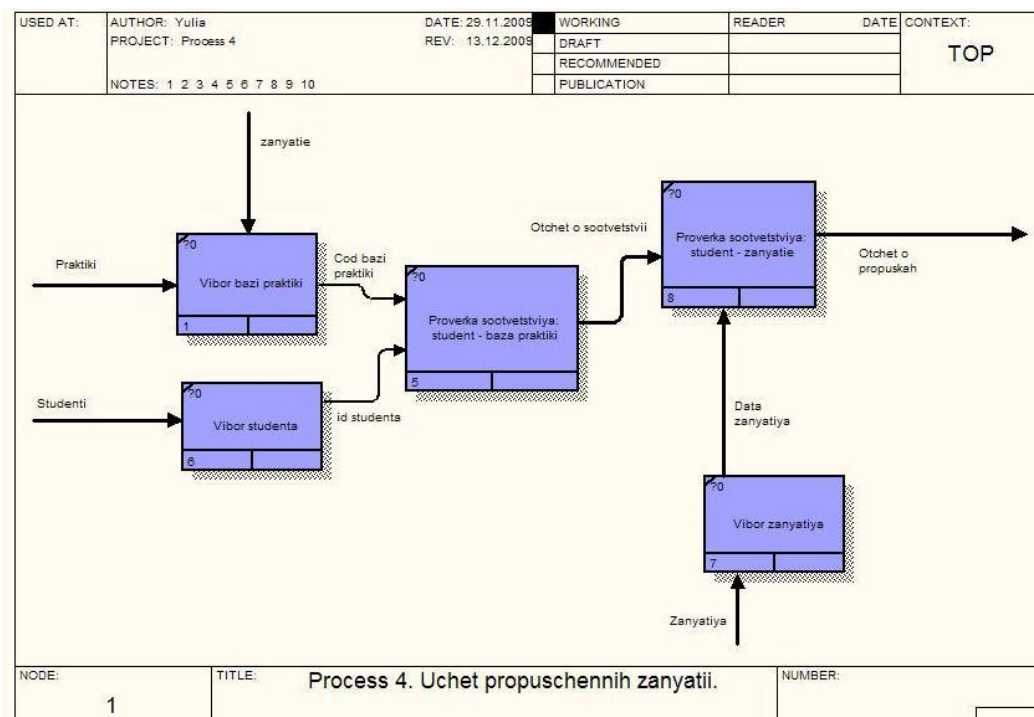
Диаграммы моделей функций процесса распределения студентов на преддипломную практику



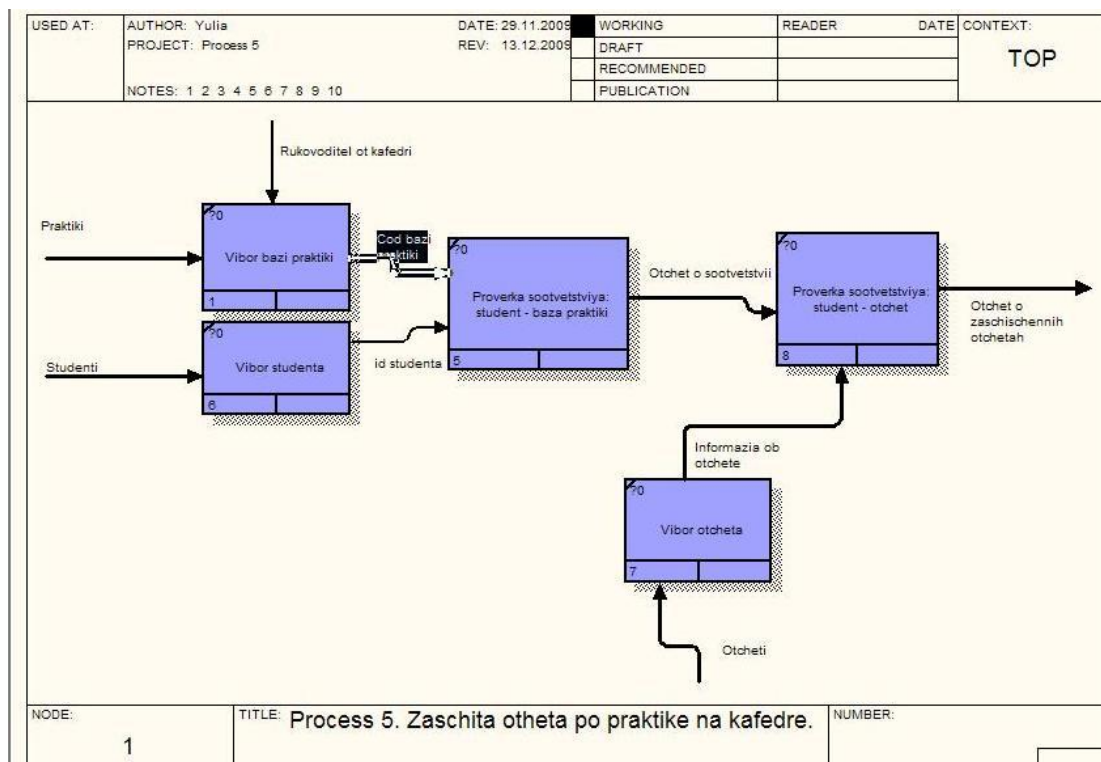
Модель функции «Назначение руководителя практики от кафедры»



Модель функции «Назначение руководителя практики от предприятия»



Модель функции «Учет пропущенных занятий»



Модель функции «Контроль защиты отчета по практике на кафедре»

Приложение 2

Примеры экранных форм интерфейса с БД

The screenshot shows a window titled "PracticeDataManipulation". It contains the following fields:

- Base name:
- Base address:
- Professor:
- Manager:

Below these fields is a table with two columns: "FIO" and "Chesked".

FIO	Chesked
Елизавета Витальевна Недачина	<input type="checkbox"/>

At the bottom of the window are "Ok" and "Cancel" buttons.

Форма поиска студентов, не распределенных на практику

The screenshot shows a window titled "PracticeDataManipulation". It contains the following fields:

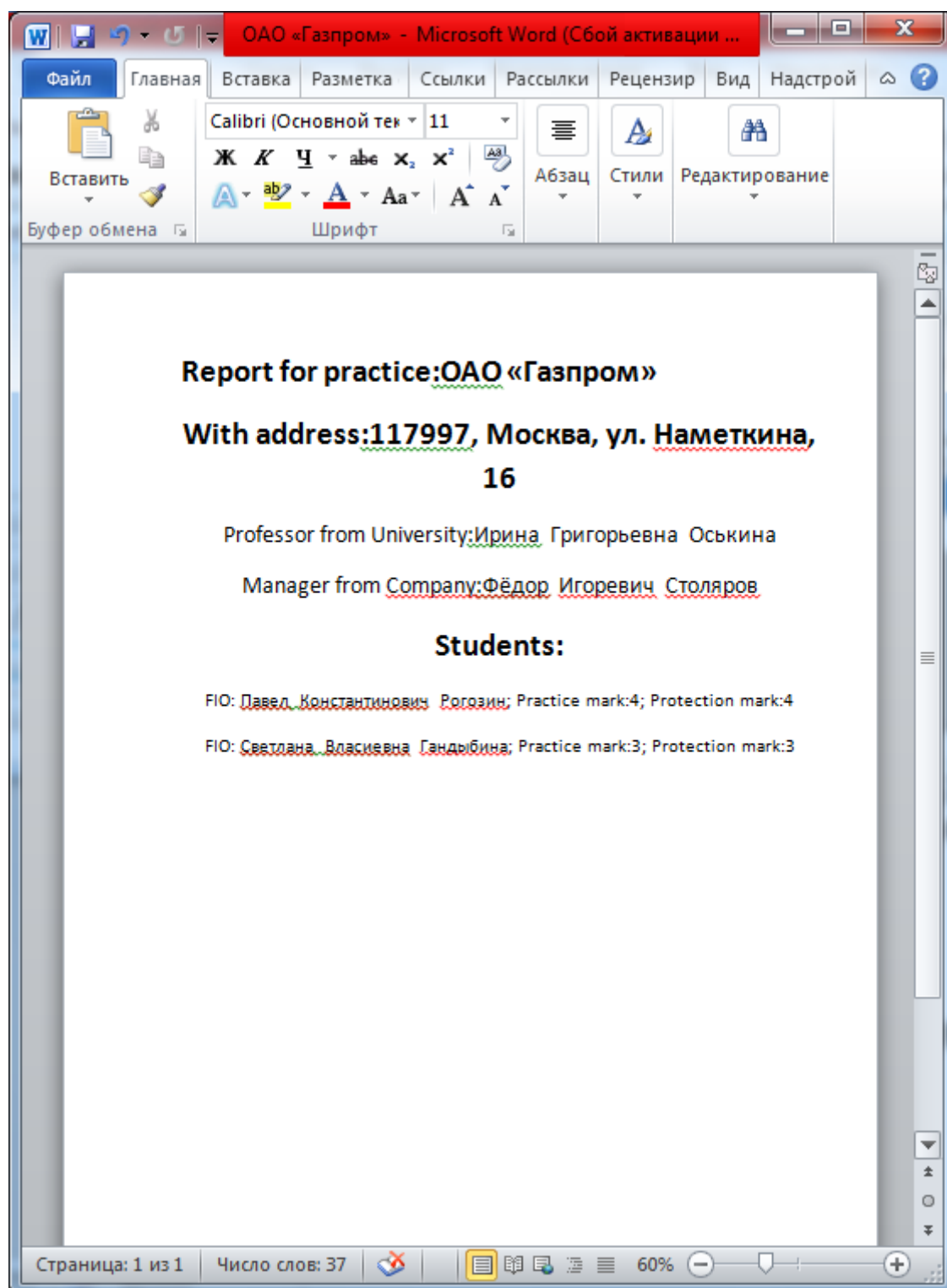
- Base name:
- Base address:
- Professor:
- Manager:

Below these fields is a table with two columns: "FIO" and "Chesked".

FIO	Chesked
Павел Констан...	<input checked="" type="checkbox"/>
Светлана Влас...	<input checked="" type="checkbox"/>
Елизавета Вит...	<input type="checkbox"/>

At the bottom of the window are "Ok" and "Cancel" buttons.

Форма поиска студентов на базе практики



Пример сформированного отчета по базе практики в формате документа Word