

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МОЭВМ

ИНДИВИДУАЛЬНОЕ ДОМАШНЕЕ ЗАДАНИЕ
по дисциплине «Введение в нереляционные базы данных»
Тема: Электронный журнал успеваемости

Студенты гр. 6383

Азаревич А.Д.

Гомонова А.А.

Студентка гр. 6382

Зацепилова С.В.

Преподаватель

Заславский М.М.

Санкт-Петербург

2019

ЗАДАНИЕ

Студенты

Азаревич А.Д.

Гомонова А.А.

Зацепилова С.В.

Группы 6383, 6382

Тема проекта: Электронный журнал успеваемости Neo4j

Исходные данные:

Необходимо реализовать приложение электронный журнал успеваемости для СУБДNeo4j.

Содержание пояснительной записки:

«Содержание»

«Введение»

«Сценарии использования»

«Модель данных»

«Разработанное приложение»

«Заключение»

«Список использованных источников»

«Приложение»

Предполагаемый объем пояснительной записки:

Не менее 15 страниц.

Дата выдачи задания:

Дата сдачи реферата:

Дата защиты реферата:

Студенты гр.6383

Азаревич А.Д.

Гомонова А.А.

Студентка гр.6382

Зацепилова С.В.

Преподаватель

Заславский М.М.

АННОТАЦИЯ

В рамках данного курса предполагалось разработать какое-либо приложение в команде на одну из поставленных тем. Была выбрана тема электронный журнал ведомости для СУБД Neo4j.

Приложение электронный журнал позволит добавлять и удалять студентов, преподавателей, ставить оценку студенту, добавлять и удалять предметы и направление обучения, назначать предметы преподавателем. С помощью приложения можно получить следующие статистики: «Средняя оценка на факультете», «Оценки, выставяемые преподавателем», «Загруженность преподавателей на кафедре».

SUMMARY

As part of this course, it was supposed to develop any application in the team on one of the topics posed. The topic was selected as the electronic journal sheet for the Neo4j DBMS.

The electronic journal application will allow you to add and remove students, teachers, rate the student, add and remove subjects and the direction of study, assign subjects as a teacher. Using the application, you can get the following statistics: "Average grade at the faculty", "Grades given by the teacher", "Workload of teachers in the department."

Содержание

ВВЕДЕНИЕ	5
1. СЦЕНАРИЙ ИСПОЛЬЗОВАНИЯ	6
1.1. Макеты пользовательского интерфейса	6
1.2. Описание сценариев использования	8
1.2.1. Просмотр главной страницы, импорт и экспорт БД	8
1.2.2. Переход на страницу с информацией о преподавателях и предметах, выполнение запросов на странице	9
1.2.3. Переход на страницу с информацией о факультетах, выполнение запросов на странице	10
1.2.4. Переход на страницу с информацией о кафедрах, выполнение запросов на странице	10
1.2.5. Переход на страницу с информацией о студентах, выполнение запросов на странице	11
1.2.6. Переход на страницу статистических запросов, выполнение запросов	12
2. МОДЕЛЬ ДАННЫХ	13
2.1. NoSQL модель данных (Neo4j)	13
2.1.1. Графическое представление	13
2.1.2. Описание назначений коллекций, типов данных и сущностей	13
2.1.3. Расчёт объема	15
2.1.4. Примеры запросов	16
2.2. SQL модель данных	16
2.2.1. Графическое представление	16
2.2.2. Описание назначений коллекций, типов данных и сущностей	16
2.2.3. Расчёт объёма	18
2.2.4. Примеры запросов	19
2.3. Сравнение Neo4j и SQL моделей данных	20
3. РАЗРАБОТАННОЕ ПРИЛОЖЕНИЕ	20
3.1. Краткое описание	20
3.2. Схема экранов приложения	21
3.3. Используемые технологии	22
ЗАКЛЮЧЕНИЕ	23
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	24
ПРИЛОЖЕНИЕ А. ДОКУМЕНТАЦИЯ ПО СБОРКЕ И РАЗВЕРТЫВАНИЮ ПРИЛОЖЕНИЯ	25

ВВЕДЕНИЕ

Цель работы – создать приложение электронный журнал успеваемости, с помощью которого можно вести учет успеваемости каждого студента, выполнять различные запросы с целью получения необходимой информации.

Было решено разработать веб-приложение, которое позволит хранить информацию о факультетах, группах, студентах, преподавателях и оценках, полученных студентами.

Качественные требования к решению: требуется разработать приложение с использованием СУБД Neo4j.

1. СЦЕНАРИЙ ИСПОЛЬЗОВАНИЯ

1.1.Макеты пользовательского интерфейса

Главная страница представлена на рис. 1, страница с информацией о студентах на рис. 2, рис. 3, страница с информацией о преподавателях на рис. 4, страница статистических запросов представлена на рис. 5, страница факультетов на рис. 6, страница кафедр представлена на рис. 7.

СПбГЭТУ "ЛЭТИ"

Information about students
Information about teachers and subjects
Statistical queries
Faculties
Departments

Import Export
Load json-file DB
Select file File not selected Export DB
Import file

Рисунок 1 – Главная страница приложения

Студенты

Применить фильтр				
Факультет	Направление	Группа	Фамилия Имя	
ФКТИ	Прикладная математика и информатика	6382	Жигунов Сергей	Отчислить
ФКТИ	Прикладная математика и информатика	6382	Глазков Дмитрий	Отчислить
ФКТИ	Прикладная математика и информатика	6382	Воропаев Андрей	Отчислить
ФКТИ	Прикладная математика и информатика	6382	Вайгачев Андрей	Отчислить
ФКТИ	Прикладная математика и информатика	6382	Тимофеев Дмитрий	Отчислить
ФКТИ	Прикладная математика и информатика	6382	Терещенко Виктория	Отчислить
ФКТИ	Прикладная математика и информатика	6382	Спас Артём	Отчислить
ФКТИ	Прикладная математика и информатика	6382	Михеева Екатерина	Отчислить
ФКТИ	Прикладная математика и информатика	6382	Михайлов Юрий	Отчислить
ФКТИ	Прикладная математика и информатика	6382	Мелвелев Глеб	Отчислить
ФКТИ	Прикладная математика и информатика	6383	Быков Илья	Отчислить
ФКТИ	Прикладная математика и информатика	6383	Василенко Татьяна	Отчислить
ФКТИ	Прикладная математика и информатика	6383	Глушков Николай	Отчислить
ФКТИ	Прикладная математика и информатика	6383	Базаров Ильмурат	Отчислить
ФКТИ	Прикладная математика и информатика	6383	Азаревич Артём	Отчислить
ФКТИ	Прикладная математика и информатика	6383	Карамышев Альберт	Отчислить
ФКТИ	Прикладная математика и информатика	6383	Гомонова Анастасия	Отчислить
ФКТИ	Прикладная математика и информатика	6383	Любчук Денис	Отчислить
ФКТИ	Прикладная математика и информатика	6383	Липендин Антон	Отчислить
ФКТИ	Прикладная математика и информатика	6383	Лавренкова Екатерина	Отчислить
ФКТИ	Прикладная математика и информатика	6381	Серелкин Дмитрий	Отчислить
ФКТИ	Прикладная математика и информатика	6381	Мартыненко Павел	Отчислить
ФКТИ	Прикладная математика и информатика	6381	Овчинников Евгений	Отчислить

Рисунок 2- Страница с информацией о студентах 1

ФКТИ	Программная инженерия	5304	Вергейчик Глеб	Отчислить
ФКТИ	Программная инженерия	5304	Герасимова Дарина	Отчислить
ФКТИ	Программная инженерия	5304	Афийчуз Иван	Отчислить
ФКТИ	Программная инженерия	5304	Вероха Вероника	Отчислить
ФКТИ	Программная инженерия	5304	Шарыпина Дарья	Отчислить
ФКТИ	Программная инженерия	5304	Швайко Тимофей	Отчислить
ФКТИ	Программная инженерия	5304	Черкасова Елена	Отчислить
ФРТ	Радиотехнические системы	4305	Нестерова Евгения	Отчислить
ФРТ	Радиотехнические системы	4305	Ширяев Ярослав	Отчислить
ФРТ	Радиотехнические системы	4305	Фналковский Максим	Отчислить
ФРТ	Радиотехнические системы	4305	Токун Михаил	Отчислить
ФРТ	Радиотехнические системы	4305	Шевелева Анна	Отчислить
ФРТ	Радиотехнические системы	4305	Шарипова Ранагуль	Отчислить
ФРТ	Радиотехнические системы	4305	Румянцев Владислав	Отчислить
ФРТ	Радиотехнические системы	4305	Попов Никита	Отчислить
ФРТ	Радиотехнические системы	4305	Степанова Софья	Отчислить
ФРТ	Радиотехнические системы	4305	Сергухин Владислав	Отчислить
ФРТ	Радиотехнические системы	4305	Лопатина Анастасия	Отчислить
ФРТ	Радиотехнические системы	4305	Поляков Нестор	Отчислить
ФРТ	Радиотехнические системы	4305	Мейзер Денис	Отчислить

Группа	Фамилия	Имя	Телефон	Город
4305 ▼	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="button" value="Добавить"/>				

Рисунок 3 - Страница с информацией о студентах2

Преподаватели

Кафедра	Фамилия	Имя	
БЖД	Овдиенко Евгений		<input type="button" value="Уволить"/>
БЖД	Трусов Александр		<input type="button" value="Уволить"/>
МОЭВМ	Заславский Марк		<input type="button" value="Уволить"/>
МОЭВМ	Клионский Дмитрий		<input type="button" value="Уволить"/>
МОЭВМ	Вербицкая Екатерина		<input type="button" value="Уволить"/>
МОЭВМ	Балтрашевич Владимир		<input type="button" value="Уволить"/>
МОЭВМ	Рясков Ян		<input type="button" value="Уволить"/>
ИБ	Племянников Александр		<input type="button" value="Уволить"/>

Кафедра	Фамилия	Имя	Телефон	Город
МОЭВМ ▼	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="button" value="Добавить"/>				

Рисунок 4 – Страница с информацией о преподавателях

ЗАПРОСЫ

Средняя оценка на факультете	Оценки, выставляемые преподавателем	Загруженность преподавателей на кафедре
<input type="button" value="На гл.страницу"/>		

Рисунок 5–Страница статистических запросов

Факультеты

ФРТ 1

ФКТИ 2

На гл.страницу

Рисунок 6 - Страница факультетов

Кафедры

МОЭВМ 4

ИБ 2

БЖД 1

На гл.страницу

Рисунок 7 - Страница кафедр

1.2.Описание сценариев использования

1.2.1. Просмотр главной страницы, импорт и экспорт БД

№	Наименование	Описание сценария	Ожидаемый результат
1	Просмотр главной страницы	Открыть главную страницу	Отображаются 8 кнопок: 1) Кнопка для перехода на страницу информации о студентах; 2) Кнопка для перехода на страницу с информацией о преподавателях и предметах; 3) Кнопка для перехода на страницу статистических запросов; 4) Кнопка для перехода на страницу факультетов; 5) Кнопка для перехода на страницу кафедр; 6) Кнопка выбора файла с пользовательского компьютера; 7) Кнопка «Импортировать файл» для для осуществления операции импорта БД; 8) Кнопка «Экспорт БД» для осуществления операции экспорта БД.
2	Импорт данных	В окошке импорта/экспорта нажать на кнопку «Выберите файл»	Открывается окно выбора файла с пользовательского компьютера.

		Выбрать файл на пользовательском компьютере, нажать на кнопку «Открыть».	Окно выбора файла с пользовательского файла закрылось, выбранный файл отображается в окошке импорта.
		Нажать на кнопку «Импортировать файл»	Данные импортированы.
3	Экспорт данных	В окошке импорта/экспорта нажать на кнопку «Экспорт БД»	Выполнено скачивание файла данных в формате .json.

1.2.2. Переход на страницу с информацией о преподавателях, выполнение запросов на странице

№	Наименование	Описание сценария	Ожидаемый результат
1	Переход на страницу с информацией о преподавателях	На главной странице нажать на кнопку «Информация о преподавателях и предметах»	Открыта страница с информацией о преподавателях и предметах, доступны кнопки «Уволить», «Добавить», «На главную страницу».
2	Увольнение преподавателя	Выбрать любого преподавателя и нажать на кнопку «Уволить»	Выбранный преподаватель не отображается в списке преподавателей.
3	Добавление преподавателя	Нажать на выпадающий список «Кафедры», выбрать необходимую кафедру	В поле «Кафедра» отображается выбранная кафедра.
		Заполнить поля «Фамилия», «Имя», «Телефон», «Город», нажать на кнопку «Добавить»	В списке преподавателей отображается добавленный преподаватель, справа от преподавателя доступна кнопка «Уволить».
4	Просмотр информации о нагрузке преподавателя	Нажать на Фамилию Имя преподавателя	Открыта страница с информацией о нагрузке преподавателя, доступны кнопки «Добавить нагрузку», «Освободить нагрузку», «На гл. страницу», «Назад».
5	Удаление/добавление нагрузки преподавателя	Нажать на кнопку «Освободить» в строке предмета, который необходимо снять с преподавателя	Предмет в таблице не отображается.
		В выпадающем списке выбрать предмет, который необходимо добавить, нажать на кнопку «Добавить нагрузку»	Выбранный предмет отображается в таблице предметов, которым обучает преподаватель.
6	Переход на главную страницу	Нажать на кнопку «На гл. страницу»	Выполнен переход на главную страницу.

1.2.3. Переход на страницу с информацией о факультетах, выполнение запросов на странице

№	Наименование	Описание сценария	Ожидаемый результат
1	Переход на страницу с информацией о факультетах	На главной странице нажать на кнопку «Факультеты»	Открыта страница «Факультеты», на странице отображаются существующие факультеты, доступна кнопка «На гл. страницу».
2	Переход на страницу с информацией о выбранном факультете	Нажать на наименование факультета	Открыта страница с информацией о факультете, отображается список направлений на выбранном факультете и количество групп. Доступны кнопки «Добавить направление», «На гл. страницу», «Назад».
3	Добавление направления	В поле «Добавить направление» ввести наименование направления, нажать на кнопку «Добавить направление»	Добавленное направление отображается в списке направлений, справа от направления доступна кнопка «Убрать».
4	Просмотр информации о направлении	Нажать на наименование направления	Открыта страница с информацией о выбранном направлении, доступны кнопки «Добавить группу», «На гл. страницу», «Назад».
5	Добавление группы	В поле «Добавить группу» ввести номер группы, которую необходимо добавить, нажать на кнопку «Добавить группу»	Добавленная группа отображается в списке групп, справа от группы доступна кнопка «Убрать».
6	Удаление группы	Справа от группы нажать на кнопку «Убрать»	Добавленная группа в списке групп не отображается.
7	Удаление направления	Справа от добавленного направления нажать на кнопку «Убрать»	Добавленное направление в списке направлений не отображается.
8	Переход на страницу с информацией о факультетах	Нажать на кнопку «Назад»	Открыта страница «Факультеты», на странице отображаются существующие факультеты, доступна кнопка «На гл. страницу».

1.2.4. Переход на страницу с информацией о кафедрах, выполнение запросов на странице

№	Наименование	Описание сценария	Ожидаемый результат
1	Переход на страницу с информацией о кафедрах	На главной странице нажать на кнопку «Кафедры»	Открыта страница «Кафедры», на странице отображаются существующие кафедры, кнопка «На гл. страницу».
2	Переход на страницу с информацией о выбранной	Нажать на наименование кафедры	Открыта страница с информацией о выбранной кафедре, отображается список направлений, соответствующие направлению предметы и количество

	кафедре		связей у каждого предмета, доступна кнопка «Добавить предмет».
3	Добавление предмета	В поле «Добавить предмет» ввести наименование предмета и нажать на кнопку «Добавить предмет»	Предмет отображается в списке предметов, справа от наименования предмета доступна кнопка «Убрать».
4	Удаление предмета	Нажать на кнопку «Убрать»	Выбранный предмет в списке предметов не отображается.
5	Переход на страницу с информацией о кафедрах	Нажать на кнопку «Назад»	Открыта страница «Кафедры», на странице отображаются существующие кафедры, кнопка «На гл. страницу».

1.2.5. Переход на страницу с информацией о студентах, выполнение запросов на странице

№	Наименование	Описание сценария	Ожидаемый результат
1	Переход на страницу с информацией о студентах	На главной странице нажать на кнопку «Информация о студентах»	Открыта страница с информацией о студентах, отображается список студентов со следующей информацией: «Факультет», «Направление», «Группа», «Фамилия Имя», доступна кнопка «Применить фильтр», «Добавить», «На гл. страницу», «Отчислить».
2	Добавление студента	В выпадающем списке выбрать группу, ввести Фамилию, Имя, телефон, город, нажать на кнопку «Добавить»	Добавленный студент отображается в списке студентов, справа от студента отображается кнопка «Отчислить».
3	Удаление студента	Справа от студента нажать на кнопку «Отчислить»	Добавленный студент в списке не отображается.
4	Фильтр по факультетам	Нажать на кнопку «Факультет» Выбрать факультет(-ы), по которому(-ым) требуется применить фильтр, нажать на кнопку «Применить фильтр»	Отображается список факультетов. В списке отображаются студенты, обучающиеся на выбранном факультете(-ах).
5	Фильтр по направлениям	Нажать на кнопку «Направление» Выбрать направление(-я), по которому(-ым) требуется применить фильтр, нажать на кнопку «Применить фильтр»	Отображается список направлений. В списке отображаются студенты, обучающиеся на выбранном направлении(-ях).
6	Фильтр по группе	Нажать на кнопку «Группа» Выбрать группу(-ы), студенты которой(-ых) должны отобразиться,	Отображается список групп. В списке отображаются студенты выбранной(-ых) группы (групп).

		нажать на кнопку «Применить фильтр»	
7	Просмотр зачетки студента	Нажать на Фамилию Имя студента	Открыта страница с информацией о студенте, отображается информация об оценках студента по каждому предмету, доступны кнопки «Удалить отметку», «Выставить».
8	Удаление оценки	Нажать на кнопку «Удалить отметку» напротив выбранного предмета	Строка с оценкой по предмету удалена.
9	Выставление оценки	Заполнить поля с информацией о предмете, преподавателе, дата сдачи и оценка, нажать на кнопку «Выставить»	Отображается строка в таблице успеваемости с соответствующей информацией о выставленной оценке.

1.2.6. Переход на страницу статистических запросов, выполнение запросов

№	Наименование	Описание сценария	Ожидаемый результат
1	Переход на страницу статистических запросов	На главной странице нажать на кнопку «Статистические запросы»	Отображаются доступные запросы: «Средняя оценка на факультете», «Оценки, выставяемые преподавателем», «Загруженность преподавателей на кафедре».
2	Запрос «Средняя оценка на факультете»	На странице статистических запросов нажать на кнопку «Средняя оценка на факультете»	Открыта страница запроса, доступен выбор параметров запроса и кнопка «Выполнить».
		Заполнить необходимые параметры запроса, нажать на кнопку «Выполнить»	Отображается результат запроса.
3	Запрос «Оценки, выставяемые преподавателем»	На странице статистических запросов нажать на кнопку «Оценки, выставяемые преподавателем»	Открыта страница запроса, доступен выбор параметров запроса и кнопка «Выполнить».
		Заполнить необходимые параметры запроса, нажать на кнопку «Выполнить»	Отображается результат запроса.
4	Запрос «Загруженность преподавателей на кафедре»	На странице статистических запросов нажать на кнопку «Загруженность преподавателей на кафедре»	Открыта страница запроса, доступен выбор параметров запроса и кнопка «Отправить запрос».
		Заполнить необходимые параметры запроса, нажать на кнопку «Отправить запрос»	Отображается результат запроса.

2. МОДЕЛЬ ДАННЫХ

2.1. NoSQL модель данных (Neo4j)

2.1.1. Графическое представление

Графическое представление модели данных Neo4j показано на рис. 8.

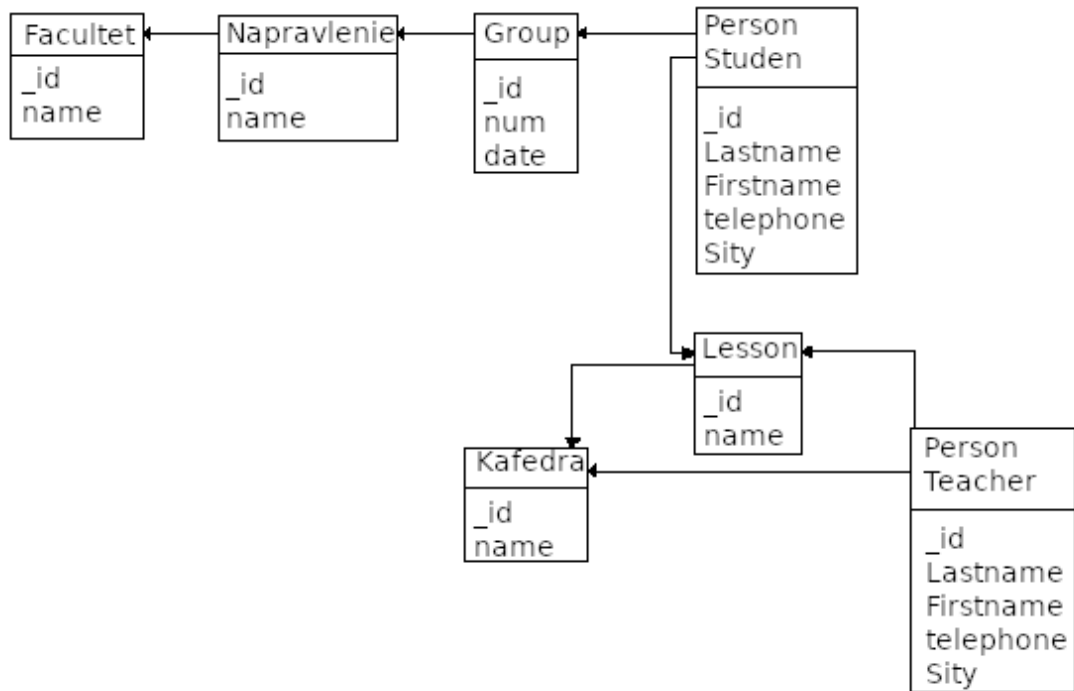


Рисунок 8 – Графическое представление модели данных Neo4j

2.1.2. Описание назначений коллекций, типов данных и сущностей

Модель состоит из 6 сущностей:

1. Факультет:

- "_id", int - идентификатор. 4B
- "name", String - название факультета. 4*2B

Итого: 12B

2. Направление обучения:

- "_id", int - идентификатор. 4B
- "name", String - Название направления. 15*2B

Итого: 34B

3. Группа:

- "_id", int - идентификатор. 4B
- "num", String - Номер группы. 4*2B

Итого: 12B

4. Человек (может быть как студентом, так и преподавателем):

- "_id", int - идентификатор. 4B
- "Lastname", String - Фамилия. 8*2B
- "Firstname", String - Имя. 6*2B
- "telephone", String - Номер телефона. 14*2B
- "Sity", String - Город. 10*2B;

Итого: 80B

5. Кафедра:

- "_id", int - идентификатор. 4B
- "name", String - Название кафедры. 4*2B

Итого: 12B

6. Предмет:

- "_id", int - идентификатор. 4B
- "name", String - Название предмета. 7*2B

Итого: 18B

Существующие связи между сущностями:

- Направление->Факультет ("Данной специальности учат на данном факультете"). Полей нет.
- Группа->Направление ("Данная группа обучается по этому направлению"). Полей нет.
- Человек(студент)->Группа ("Студент зачислен в данную группу"). Полей нет.
- Человек(Студент)->Предмет ("Студент имеет в зачётке отметку по предмету"). Поля:
 - "_id", int - идентификатор. 4B
 - "assessment", char - Полученная оценка. 2B

- "date", String - Дата выставления оценки. $10 \times 2B$
- "teach_id", int - Идентификатор преподавателя. $4B$

Итого: 30B

- Человек(Преподаватель)->Кафедра ("Преподаватель работает на кафедре"). Полей нет.
- Человек(Преподаватель)->Предмет ("Преподаватель ведёт"). Полей нет.
- Предмет->Кафедра ("Предмет изучается на кафедре"). Полей нет.

2.1.3. Расчёт объема

Предположим, что существует F факультетов, на каждом факультете происходит обучение в среднем по 4-м направлениям, на каждом направлении сформировано в среднем по 3 группы, в каждой из которых в среднем обучается по 15 человек, каждый из которых сдал по 3 предмета. Так же предположим, что есть на каждом факультете присутствует в среднем по 3 кафедры, на которых в среднем преподают по 5 преподавателей. На каждой кафедре в среднем проходят 3 предмета.

"Чистый" объём: Объём сущности "Человек" в среднем составляет 80B, сущностей "Факультет", "Группа" и "Кафедра" - 12B, сущности "Направление" - 34B, сущности "Предмет" - 18B. Объём связи "Студент->Предмет" - 30B. Тогда занимаемый объём: $32290 \times F$ байт.

Фактический объём: Объём сущностей:

- "Человек" - 817B
- "Факультет" - 216B
- "Направление" - 503B
- "Группа" - 784B
- "Предмет" - 638B
- "Кафедра" - 216B

Связь "Студент->Предмет" - 713В

Тогда общий объём БД: $558533 * F$ байт

Избыточность модели: $558533 / 32290 = 17.3$

2.1.4. Примеры запросов

1. Добавить нового преподавателя

```
MATCH (k:Kafeda{name:"Название"}),  
(l:Lesson{name:"Название"}) CREATE  
(p:Person:Teacher{...})-[:work]-(k), (p)-[:teach]->(l);
```

2. Определить средний бал по предмету "HN" для каждой группы (которая его проходила)

```
match (g)--()-[r]-(:Lesson{name:"HN"}) return g.num,  
avg(toInteger(r.assessment)) as q;
```

2.2. SQL модель данных

2.2.1. Графическое представление

Графическое представление модели данных SQL показано на рис. 9.

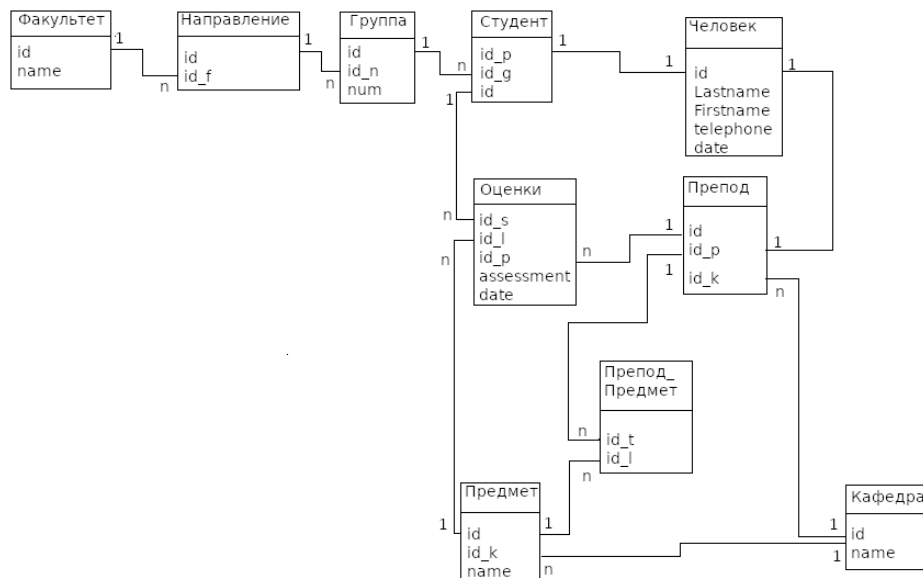


Рисунок 9 - Графическое представление модели данных SQL

2.2.2. Описание назначений коллекций, типов данных и сущностей

SQL модель состоит из следующих таблиц:

1. Факультет:

- id, int - уникальный номер записи. 4B
- "name", String - название факультета. 4*2B

Итого: 12B

2. Направление обучения:

- "_id", int - идентификатор. 4B
- "id_f", int - "ссылка" на факультет. 4B
- "name", String - Название направления. 15*2B

Итого: 38B

3. Группа:

- "_id", int - идентификатор. 4B
- "id_n", int - "ссылка" на факультет. 4B
- "num", String - Номер группы. 4*2B

Итого: 16B

4. Человек:

- "_id", int - идентификатор. 4B
- "Lastname", String - Фамилия. 8*2B
- "Firstname", String - Имя. 6*2B
- "telephone", String - Номер телефона. 14*2B
- "Sity", String - Город. 10*2B;

Итого: 80B

4.1 Студент:

- "_id", int - идентификатор. 4B
- "id_p", int - "ссылка" на человека. 4B

Итого 8B

4.2 Преподаватель:

- "_id", int - идентификатор. 4B
- "id_p", int - "ссылка" на человека. 4B

ИтогоЖ 8B

5. Кафедра:

- "_id", int - идентификатор. 4B

- "name", String - Название кафедры. 4*2B

Итого: 12B 6. Предмет:

- "_id", int - идентификатор. 4B
- "id_k", int - на какой кафедре изучается. 4B
- "name", String - Название предмета. 7*2B

Итого: 22B 7. Препод_Предмет:

- "id_t", int - какой препод. 4B
- "id_l", int - какой предмет ведёт. 4B

Итого: 8B 8. Оценки:

- "id_s", int - Студент. 4B
- "id_l", int - по предмету. 4B
- "id_p", int - преподаватель. 4B
- "assessment", char - поставил отметку. 2B
- "date", String - такого числа. 10*2B

Итого: 34B

2.2.3. Расчёт объёма

Предположим, что существует F факультетов, на каждом факультете происходит обучение в среднем по 4-м направлениям, на каждом направлении сформировано в среднем по 3 группы, в каждой из которых в среднем обучается по 15 человек, каждый из которых сдал по 3 предмета. Так же предположим, что есть на каждом факультете в среднем есть по 3 кафедры, на которых в среднем преподают по 5 преподавателей. На каждой кафедре в среднем проходят 3 предмета. В среднем, каждый преподаватель ведёт по 2 предмета.

"Чистый" объём: Средний размер записи для каждой таблицы (в байтах):

- "Факультет" – 12
- "Направление" – 38
- "Группа" – 16
- "Человек" – 80
- "Студент" – 8

- "Преподаватель" – 8
- "Кафедра" – 12
- "Предмет" – 22
- "Отметка" – 34
- "ПрепоДПредмет" – 4

Тогда общий объём БД: $36350 * F$ байт

Фактический объём: Объём записей в таблице (в байтах):

- "Факультет" – 292
- "Направление" – 300
- "Группа" – 297
- "Человек" – 308
- "Студент" – 288
- "Преподаватель" – 288
- "Кафедра" – 288
- "Предмет" – 293
- "Отметка" – 318
- "ПрепоДПредмет" – 287

Тогда общий объём БД: $299044 * F$ байт

Избыточность модели: $299044 / 36350 = 8.2$

2.2.4. Примеры запросов

1. Добавить нового преподавателя

```
INSERT INTO Человек VALUES (...)
INSERT INTO Преподаватель VALUES (...)
INSERT INTO ПрепоДПредмет VALUES (...)
```

2. Определить средний бал по предмету "HN" для каждой группы (которая его проходила)

```
SELECT num, avg(Assessment) AS q FROM ( SELECT num,
id_gr, Студенты.id as qq FROM Группы LEFT JOIN
Студенты ON Группы.id = Студенты.id_g ) INNER JOIN
Оценки ON qq=Оценки.id_s GROUP BY id_gr
```

2.3. Сравнение Neo4j и SQL моделей данных

Запросы на SQL-языке являются более громоздкими, чем на языке Neo4j Cypher. В реализации реляционной модели данных пришлось бы создать слишком большое количество таблиц для связи всей информации.

Neo4j-БД требует больше памяти по сравнению с SQL-БД для хранения полей. Однако время выполнения запроса в Neo4j-БД в среднем составляет 57 мс, в то время как в SQL-БД – 108 мс.

Исходя из вышеперечисленного, можно сказать, что для рассматриваемой задачи Neo4j подходит в большей мере.

3. РАЗРАБОТАННОЕ ПРИЛОЖЕНИЕ

3.1. Краткое описание

Для упрощения процесса разработки было принято решение разделить приложение на клиентскую и серверную части.

Клиентская часть реализована с использованием JavaScript, HTML, шаблонизатора Pug и JS библиотек JQuery и AnyChart.

Серверная часть приложения реализована с использованием Node.js и фреймворка express. Так же использовались следующие фреймворки:

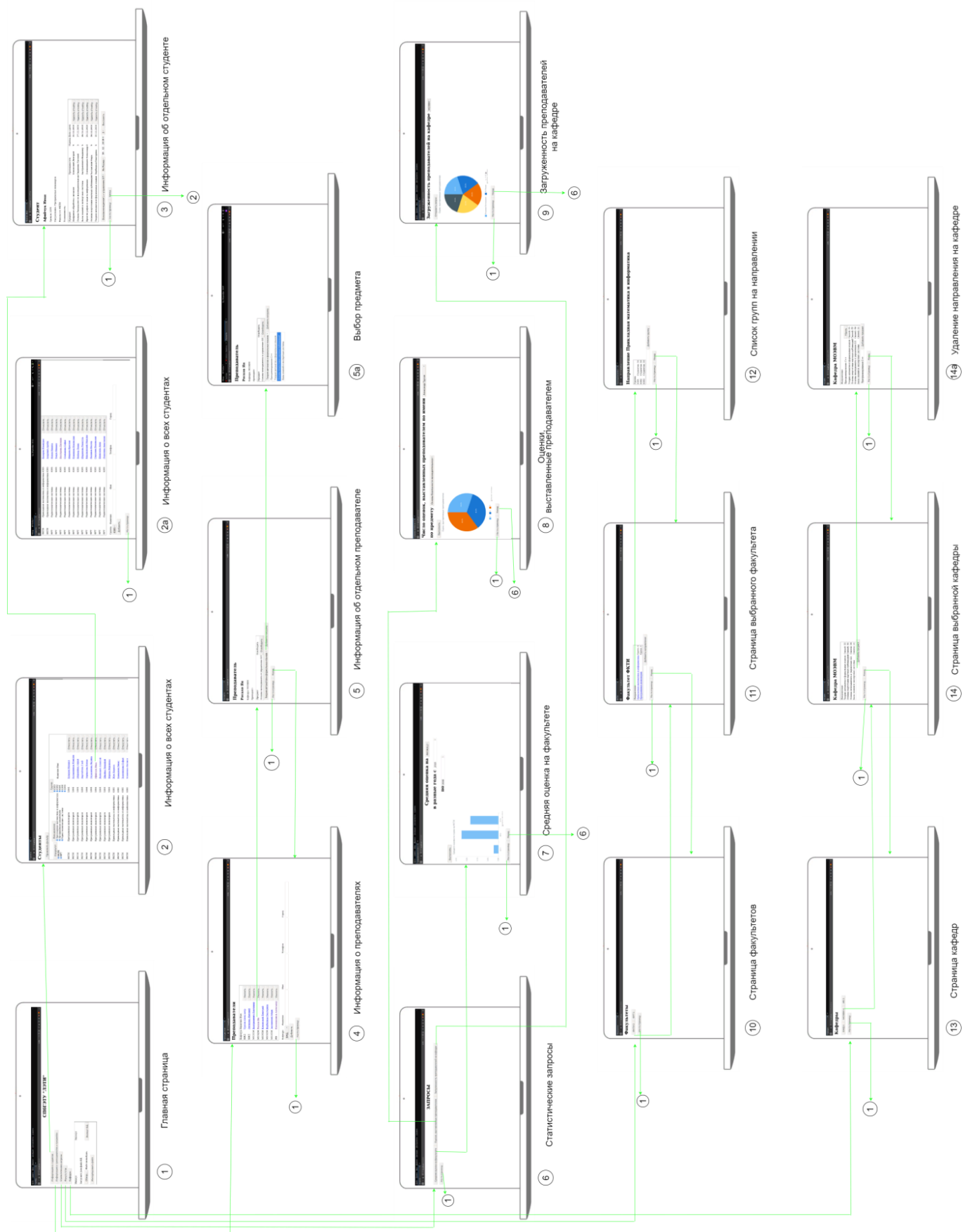
- pug (для генерации HTML-страниц по заданным шаблонам);
- body-parser (для работы с клиентскими запросами), multer (для возможности загрузки файлов с импортируемыми данными от клиентской части);
- neo4j-driver (для работы с базой данных Neo4j).

Поскольку приложение требует установки стороннего программного обеспечения: Node.js и Neo4j – было решено для упрощения развертывания приложения использовать docker.

Ссылка на приложение доступна в разделе "Список использованных источников": [1]

3.2.Схема экранов приложения

Схема экранов приложения представлена на рис. 10.



3.3.Использованные технологии

БД: Neo4j

Backend: Node.js, express, body-parser, multer, neo4j-driver

Frontend: JS, HTML, Pug, JQuery, AnyChart

ЗАКЛЮЧЕНИЕ

В ходе работы было разработано приложение, позволяющее пользователю вести электронный журнал успеваемости, а именно: добавлять и удалять студентов, преподавателей, ставить оценку студенту, добавлять и удалять предметы и направление обучения, назначать предметы преподавателем. С помощью приложения можно получить следующие статистики: «Средняя оценка на факультете», «Оценки, выставяемые преподавателем», «Загруженность преподавателей на кафедре». Также были реализованы функции импорта и экспорта данных.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Ссылка на репозиторий: <https://github.com/moevm/nosql2h19-marks-neo4j>
2. Документация Neo4j: <https://neo4j.com/docs/cypher-manual/current/>
3. Документация Арос (экспорт):
<https://neo4j.com/docs/labs/арос/current/export/json/>
4. Документация Арос (импорт):
<https://neo4j.com/docs/labs/арос/current/import/load-json/>

ПРИЛОЖЕНИЕ А. ДОКУМЕНТАЦИЯ ПО СБОРКЕ И РАЗВЕРТЫВАНИЮ ПРИЛОЖЕНИЯ

Для использования приложения требуется наличие docker.

Установка приложения выполняется посредством команды:

```
docker pull trasil/marksdb
```

Запуск приложения выполняется с помощью команды:

```
Docker run -p 3000:3000 trasil/marksdb
```

После запуска приложение становится доступно по адресу:

```
http://localhost:3000/
```