

4. Минимизация затрат на канцелярские предметы и жесткие диски.

Преподавателю и студенту (ученику) порой приходится тратить на листы для печати, ручки и тому подобное. Поэтому в этом случае будет удобнее использовать «облачное» хранилище.

5. Исключение потери данных.

Невозможно потерять или забыть готовое задание, если вы работаете в «облаке» так как все документы, будут сохраняться сразу на сервере «облака». Потеря данных на сервере случается крайне редко.

Для проведения исследования было выбрано «облако», бесплатно предоставленное компанией Google, которое называется Google Диск. В данном «облаке» была создана лекция и презентация по теме «Интерполирование функций», которую могут использовать для самостоятельного обучения студенты образовательных учреждений.

На основании выше сказанного, можно сделать вывод о том, что «облачные» технологии предоставляют практически безграничные возможности благодаря своим сервисам, начиная с простого хранения информации и заканчивая предоставлением сложных безопасных ИТ-инфраструктур.

Главным плюсом использования «облачных» технологий в образовании является организация совместной работы между учащимися и преподавателем.

Как и в любой технологии, у «облачных» технологий есть несколько недостатков, такие как:

1. Необходимо иметь постоянный доступ в сеть Интернет. Для работы с «облаком» требуется иметь соединение с сетью Интернет.

2. Не каждая программа может быть доступна для удаленного доступа. Некоторые программы, установленные на рабочем компьютере намного функциональнее их аналогов в «облаке».

3. На сегодня недостаточно проработаны методические и технологические аспекты применения «облачных» технологий в образовательном процессе. Поэтому использование облачных технологий приходит в образование с задержкой и еще не нашло широкого применения [3]. В большинстве образовательных учреждений с недоверием относятся к использованию виртуальных «облаков» и предпочитают работать с конкретным, желательно собственным, оборудованием и с данными, которые хранятся локально [3].

**Заключение.** В результате данного исследования «облачных» технологий были рассмотрены возможные способы применения их в сфере образования, проанализированы недостатки, связанные с использованием хранилищ. Было создано приложение в «облачном» хранилище Google Диск, которое можно использовать для изучения темы «Интерполирование функций». Также была создана общая папка для студентов, в которую они могут загружать отчеты по лабораторной работе.

Литература:

1. Батура, Т. Облачные вычисления: основные понятия, задачи и тенденции развития / Т. Батура // Электронный научный журнал [Электронный ресурс]. – 2014. – №1 – Режим доступа: <http://swsys-web.ru/cloud-computing-basic-concepts-problems.html>. – Дата доступа: 05.12.2016.
2. Емельянова, О. Применение «облачных» технологий в образовании / О. Емельянова // Рос. междунар. науч. журнал. – 2014. – № 3. – С. 907–909.
3. Ратушная, Е. Облачные вычисления: новые технологии в образовании / Е. Ратушная // Международный студенческий вестник [Электронный ресурс]. – 2014. – № 1. – Режим доступа: <http://eduherald.ru/article/view?id=11820>. – Дата доступа: 05.12.2016.

## **СЕРВЕРНЫЙ МОДУЛЬ СИСТЕМЫ УВЕДОМЛЕНИЯ ОБ ИЗМЕНЕНИЯХ В РАСПИСАНИИ ЗАНЯТИЙ**

**Жданова В.С.,**

*студент ВГУ имени П.М. Машерова, г. Витебск, Республика Беларусь*

Научный руководитель – Новый В.В.

В наше время мобильные телефоны стали неотъемлемой частью жизни. Они превратились из простого средства связи в многофункциональный инструмент, без которого уже сложно представить нашу жизнь. Не надо заранее узнавать и запоминать погоду, расписание автобусов и прочее, всё это легко можно узнать с помощью мобильного телефона в любой момент времени. Но не всё ещё в этом мире представлено в виде удобных приложений. И многие вещи всё-таки приходится выяснять заранее, что может сопровождаться множеством проблем и неудобств.

Рассмотрим на примере расписания учебного учреждения, в частности, ВГУ имени П.М. Машерова.

Расписание составляется сотрудником деканата факультета, и вывешивается в холле факультета. Но периодически происходят изменения, которые должны тут же быть отображены в расписании. И возникает множество проблем, как у преподавателей, так и у студентов. Наличие подобных проблем и необходимость их решения обосновывает актуальность данной работы.

Таким образом, целью представленной работы является описание подхода к реализации серверной части клиент-серверного мобильного приложения для просмотра расписания.

**Материал и методы.** Для решения поставленной задачи используются MySQL и Workbench – для создания базы данных и хранения расписания, фреймворк Spring – для решения многих задач сервера, в том числе и облегчения связи базы данных с сервером. Для получения расписания, данные считываются из настольного приложения и отдаются на хранение в базу данных, откуда в последующем будут изъятые мобильным приложением для отображения пользователю.

**Результаты и их обсуждение.** Итогом работы является клиент-серверное приложение, которое отображает расписание для преподавателя или для студента определённой группы. Клиент и сервер представляют собой два разных приложения, взаимодействующих друг с другом. Клиентская часть приложения инициирует связь с сервером, регистрируясь в системе и отправляя информацию о пользователе в базу данных. В дальнейшем эта информация будет использоваться при обращении к базе данных с запросом о расписании. При изменении расписания, сервер будет отслеживать различия и сообщать об этом клиенту.

Предполагается, что сервер и настольное приложение находятся на разных компьютерах. Настольное приложение удалённо связывается с сервером посредством RMI (программный интерфейс вызова удаленных методов в языке Java).

Схема взаимодействия клиентской, серверной части и настольного приложения представлена на рисунке 1.

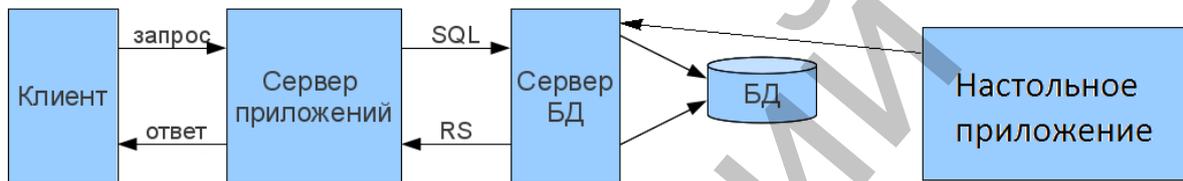


Рисунок 1 – Схема взаимодействия между компонентами приложения

В общем виде сервер имеет классическую трёхуровневую архитектуру: Клиент ↔ контроллер ↔ сервис ↔ DAO(слой объектов которые обеспечивают доступ к данным) ↔ база данных [1].

Разработка базы данных занимала значительную часть данной работы. Определение основных сущностей происходило из вопроса: кто будет пользоваться данным приложением? В итоге база данных проектируемого приложения содержит пять таблиц: student, lecturer, lesson, user и group.

Схема базы данных представлена на рисунке 2.

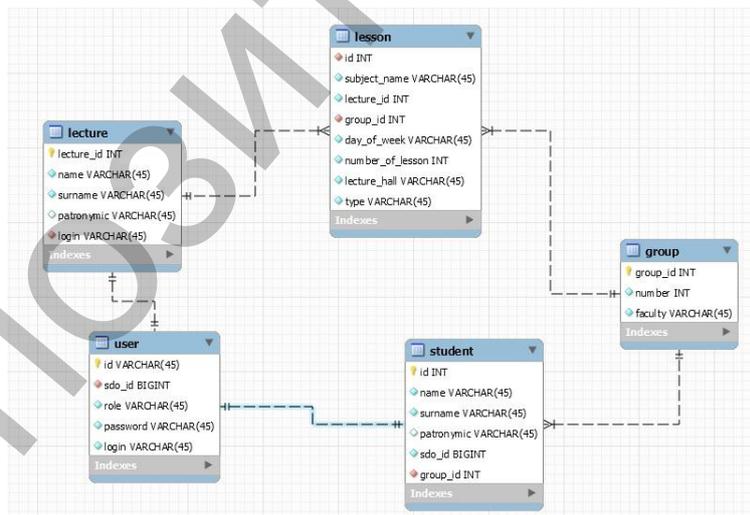


Рисунок 2 – Схема базы данных

**Заключение.** В работе рассмотрена серверная часть мобильного приложения. Так же создано тестовое приложение, демонстрирующее работоспособность сервера. В дальнейшем будет написано отдельное клиентское приложение, которое упростит получение информации студентами/преподавателями о расписании.

Литература:

1. Spring Data JPA [Электронный ресурс] / © 2017 Pivotal Software – Режим доступа: <http://docs.spring.io/spring-data/jpa/docs/1.4.3.RELEASE/reference/html/>. – Дата доступа: 01.12.2016–15.12.2016.