

РАЗРАБОТКА ПРОГРАММНЫХ СРЕДСТВ АВТОМАТИЗАЦИИ ПРОЦЕССА СОСТАВЛЕНИЯ РАСПИСАНИЯ УЧЕБНЫХ ЗАНЯТИЙ

И.А. Хмурчик, магистрант

Ж.С. Мордвинова, магистрант

*Научный руководитель – Т.Н. Изосимова, к. физ.-мат.н., доцент
Гродненский государственный университет им. Я.Купалы*

Формирование расписания занятий для учебных заведений в автоматическом режиме остается по-прежнему актуальной задачей. Появление новых компьютерных информационных технологий, методов и способов ее решения, а также более современных технических средств, приводит к необходимости менять подходы к ее программной реализации.

Разработка автоматизированной системы составления оптимального расписания, которая позволяет реализовать все возможности структурно-логических схем и обеспечивает методически правильное планирование учебной работы на семестр студентам и преподавателям, используя при этом эффективные алгоритмы и подходы, связана с определенными трудностями. При ее реализации приходится учитывать многочисленные факторы, условия их функционирования и сложные взаимосвязи. Поэтому, как правило, разработчики ограничиваются созданием компьютерных систем, при работе с которыми предпочтение принимать те или иные решения отдается пользователю. Создаваемая автором статьи автоматизированная система формирования расписания учебных занятий относится именно к такому типу разработок. Однако она содержит много новых приемов и сервисных средств, реализация которых стала возможна только за счет появившихся новых компьютерных языков и технологий.

Следует отметить, что система разрабатывается с учетом специфики составления расписания в учреждении образования («Гродненский государственный аграрный университет»), в котором его формирование сильно усложняется необходимостью проведения занятий в разных корпусах вуза, вызванное дефицитом аудиторного фонда. При этом приходится принимать во внимание близость их расположения и вместимость помещений, а также занятость преподавателей, закрепленных за определенными дисциплинами, поточность занятий и работу вуза в две смены.

Разрабатываемая авторами система имеет клиент-серверную архитектуру. Это позволяет организовать работу по составлению расписания с нескольких рабочих мест одновременно. Необходимость такого подхода объясняется тем, что формирование расписания учебных занятий по каждому факультету осуществляется закрепленным за ним работником диспетчерской службы. В связи с этим в системе предусмотрена программная поддержка следующих ролей пользователей:

- *Системный администратор* определяет права доступа пользователям, имеет возможность формирования и корректировки всей информации;
- *Администратор* отвечает за поддержку информационной базы;
- *Диспетчер* формирует расписание учебных занятий, печатает его и размещает в Интернете;
- *Администратор кафедры* генерирует и печатает расписание для преподавателей.
- *Внешний пользователь* осуществляет просмотр готового расписания.

Вход с систему происходит после ввода логина и пароля. В результате она распознает пользователя и предоставляет ему соответствующий набор функциональности.

Ниже перечисляются основные сущности системы и указываются связи между ними:

- *Факультет* содержит связь типа «один ко многим» с сущностями *Кафедра* и *Специальность*;
- *Кафедра* имеет отношение типа «один ко многим» с сущностями *Закрепление*, *Преподаватель* и *Дисциплина*;
- *Специальность* имеет связь «один ко многим» с сущностями *Группа* и *Дисциплина*;
- *Форма обучения* связана как «один ко многим» с сущностью *Специальность*;
- *Группа* представляет собой родительскую сущность с отношением «один ко многим» для сущности *Подгруппа*;
- *Корпус* имеет отношение «один ко многим» с сущностью *Аудитория*;
- *Тип аудитории* содержит связь типа «один ко многим» с сущностями *Аудитория*;
- *Закрепление* имеет отношение «один ко многим» с сущностями *Дисциплина*, *Группа*, *Преподаватель*;
- *Вид занятий* содержит связь «один ко многим» с сущностями *Закрепление*, *Тип аудитории*;
- *Занятость преподавателей* имеет связь «один ко многим» с сущностью *Преподаватель*;
- *Расписание* содержит связь «один ко многим» к таким сущностям как *Дисциплина*, *Преподаватель*, *Группа*, *Аудитория*.

В системе планируется формирование расписания учебных занятий с достаточно активным использованием человеческого фактора. Она разрабатывается с учетом автоматической проверки занятости и специфики используемых ресурсов, а также оптимизацией по следующим критериям [1]:

- количество *окон* преподавателей;
- количество *окон* учебных групп;
- проведение занятий в одном корпусе или близлежащих.

Предполагается следующий сценарий работы с системой. На первом этапе составления расписания осуществляется подготовка информационной базы, далее происходит выбор факультета, специальности, формы обучения, указывается период действия расписания. В результате автоматически генерируется область для составления расписания с учетом имеющихся для выбранной специальности и курса групп и подгрупп в рамках соответствующего факультета. Она представляет собой электронную таблицу, ячейки которой заполняются пользователем. При этом диспетчеру предлагаются только те дисциплины, которые читаются студентам в данный период времени. После выбора дисциплины и вида занятия (лекция, лабораторные, практические и т.д.) автоматически определяется фамилия преподавателя, закрепленного за данным учебным курсом. Параллельно выполняется проверка его занятости. Если читается лекция, то проверяется занятость всех остальных групп, которые обучаются на данном потоке. Далее выбирается аудитория для проведения занятия. Сначала указывается корпус, после чего пользователю предлагается список помещений, которые могут использоваться для проведения рассматриваемых занятий. Здесь учитывается и их вместимость, и особенность (например, компьютерный класс или физическая лаборатория).

Программная реализация задачи выполняется на языке Java с использованием технологий JDO (Java Data Objects)[2]. В качестве системы управления базами данных используется MySQL[3].

Несмотря на то, что система разрабатывается с учетом специфики составления расписания учебных занятий в Гродненском государственном аграрном университете, она легко может быть адаптирована для любого другого учебного заведения. Ее использование позволит сократить время на формирование расписания и повысить эффективность эксплуатации соответствующих ресурсов университета.

Список использованных источников

1. Сухарев, А. Г. Курс методов оптимизации / А. Г. Сухарев, А. В. Тимохов, В. В. Федоров М.: Наука, 1986. – 200 с.
2. Java Data Objects [Электронный ресурс]. – Режим доступа: <http://java.sun.com/products/jdo/>. – Дата доступа: 02.02.2010.
3. Веллинг, Л. Разработка web-приложений с помощью PHP и MySQL/ Л. Веллинг – Вильямс, 2004. – 410с.