

Архитектура системы управления рабочим процессом

77-48211/649649

10, октябрь 2013

Багдасарян Е. А., Королев В. С., Черненький М. В.

УДК 681.5

Россия, МГТУ им. Н.Э. Баумана

sunday@nxt.ru

snow2007@post.ru

В современном мире практически каждое предприятие, имеющее в своем штате несколько десятков сотрудников, переходит на электронные системы контроля рабочих процессов: внедряются системы технической поддержки (типа HelpDesk), системы управления взаимоотношениями с клиентами (класса CRM), системы управления бизнес процессами (известные как BPM) и другие. Такой подход позволяет существенно облегчить сотрудникам ведение рутинных операций бизнес-документирования.

Служба технической поддержки — это класс систем, позволяющих сообщать о неполадках в оборудовании инженерам, ответственным за его работу, и контролировать заказчикам выполнение работ. Основная выгода от использования такой системы заключается в аккумулировании знаний с возможностями анализа результатов работы. Например, ошибка, выявленная в одной из программ сотрудника, может выявиться и у его коллеги, и полученная ранее информация обязательно пригодиться в решении данной проблемы, даже если инженер службы поддержки не сталкивался с этой проблемой до этого лично. Анализ за определенный период всех заявок позволяет выявлять по статистическим данным наиболее проблемные средства и предпринимать превентивные меры до их массового проявления.

Система управления взаимоотношения с клиентами — это класс систем, предназначенных для автоматизации учета взаимодействий с заказчиками и клиентами, для улучшения обслуживания клиентов путём сохранения информации о клиентах и истории взаимоотношений с ними, для повышения уровня продаж и оптимизации маркетинга за счет улучшения бизнес-процессов по итогам анализа накопленных результатов. Здесь также используется типовой метод сбора и анализа статистических данных.

Система управления бизнес-процессами — это класс систем, которые позволяют полностью описать и контролировать все возможные пути следования информации от источника к

получателю. Главной их особенностью является помощь в организации внутренних бизнес-процессов, разрабатываемых руководством предприятия для того, чтобы обеспечить контроль всех этапов выполнения задач и автоматизировать трансляцию распоряжений подчиненным, а также информирования руководства о ходе выполнения работ.

Таким образом, представленные системы позволяют значительно упрощать жизнь сотрудников и позволяют им сосредотачиваться на выполнении работы, а анализ и обработка результатов будут выполняться автоматически [1].

Научная деятельность в рамках университета сопровождается существенным количеством отчетов, приказов, совещаний. Выработался определенный способ эффективного документного сопровождения научной работы. Но в условиях расширения перечня и увеличения объемов работ, наблюдаемых в настоящее время, возникает большое количество важной информации, такой как новые приказы, деловые контакты, полезные знания о конференциях, выставках и контрагентах, сведения о сопутствующих мероприятиях. В настоящее время необходимо проведение оптимизации работы университетского информационного центра сопровождения научных разработок, для того чтобы вся организационная работа не оседала в виде разрозненных документов в различных подразделениях. Сейчас для анализа результатов приходится выполнять много рутинной работы с целью поиска нужной информации, который отнимает существенное время. Отсутствие централизованной системы является ограничением в распространении, так называемых, «полезных» знаний. Информации о выставках и конференциях, отчеты о достижениях, отзывы экспертов хранятся на разобщенных компьютерах без единого доступа, а значит получить информацию о каком-либо событии можно у разных людей, что вносит определенный дискомфорт для людей, умеющих пользоваться Википедией, где на любой вопрос можно найти приемлемый ответ. Для современной организации процесса разработки и исследований каждый исследователь должен иметь быстрый доступ к необходимым данным и не заботиться об организации стандартных действий.

Целью данной работы является разработка для университета системы управления рабочим процессом (СУРП). СУРП должна позволять доводить задачи до ответственных лиц, хранить структурированную информацию, обеспечивать контролируемый доступ, управлять поставленными задачами, отображать все этапы их выполнения, создавать и взаимодействовать с событиями и обеспечивать дружественный интерфейс использования. Мы решили разбить нашу разработку на несколько этапов. Этапы различаются по функциональности, и система после завершения каждого этапа должна быть работоспособной и позволять эксплуатацию в объеме текущей реализации. Сами этапы будут описаны ниже, а для начала рассмотрим архитектуру СУРП.

Система создана по архитектурной модели «Программное обеспечение как услуга» (SaaS) [2]. SaaS — это модель использования программного обеспечения, при которой поставщик разрабатывает веб-приложение и самостоятельно управляет им, предоставляя заказчикам доступ к программному обеспечению через Интернет. Такой подход является достаточно эффективным и гибким, что подтверждает опыт использования системы «Электронный Университет» в МГТУ им. Н.Э.Баумана.

Для процесса разработки предлагаемой системы предполагается использование следующего программного обеспечения, являющегося бесплатным:

- ▲ Среда разработки Eclipse for JavaEE Developers – удобное и эффективное средство для создания приложений на языке Java, поддерживающее работу в группе, взаимодействие с сервером, режим отладки [3];
- ▲ Java EE SDK — набор Java-библиотек для разработки и запуска приложений на сервере;
- ▲ MySQL — СУБД и графический интерфейс MySQLworkbench, имеющие широкую популярность в интернет-приложениях, благодаря простоте использования и настройки;
- ▲ OpenOffice — ПО для ведения документации;
- ▲ Apache Tomcat — сервер приложений, имеющий удобный графический интерфейс и поддерживающий работу с популярными серверными технологиями;
- ▲ JQuery — библиотека JavaScript, нацеленная на взаимодействие JavaScript и HTML. Библиотека JQuery помогает обеспечить доступ к любому программному элементу архитектуры DOM, обращаться к атрибутам и содержимому элементов DOM, манипулировать ими.

Технологии и языки программирования, которые используются в нашей разработке:

- ▲ SQL — язык для создания таблиц, корректирования текущей информации, просмотра результатов в базе данных;
- ▲ сервлеты и JSP — Java-технология написания серверных компонентов с бизнес-логикой как основная рабочая часть, обеспечивающая взаимодействие с БД и обработку данных [4];
- ▲ HTML, CSS, JavaScript — технологии, позволяющие создавать, настраивать внешний вид представления, и управлять клиентской частью приложения.

Базовая архитектура системы показана на рисунке 1. На каждом этапе разработки все элементы этой схемы должны быть разработаны и протестированы. Разработка может вестись несколькими разработчиками параллельно. Работу с клиентской частью удобно вести отдельно от работы с серверной частью.



Рисунок 1. Архитектура СУРП

Данная модель разработки была выбрана из-за гибкости и простоты исполнения, а также возможности постепенного наращивания функционала в ходе получения опыта работы. Высокая гибкость и надежность в процессе разработки выгодно отличает технологию Java. Библиотека JQuery позволяет просто управлять клиентской частью веб-приложения. Также JQueryMobile позволяет ускорить создание приложения для мобильных устройств и планшетов, а такая возможность важна для будущего развития СУРП.

Рассмотрим теперь строение нашей системы более подробно. Начнем с основных функциональных элементов системы СУРП на первом этапе разработки. Это - классы и таблицы в базе данных, с которыми мы работаем:

- ♣ сотрудник (главная часть программы);
- ♣ кафедра;
- ♣ контакты (клиенты);
- ♣ задачи;
- ♣ события;
- ♣ комментарии;
- ♣ права доступа.

Все эти элементы связаны между собой отношениями, которые показаны на рисунке 2. Для наглядности на этой схеме базы данных указаны не все инфологические связи. Например, в таблицах «Комментарии», «Логирование» и «Права доступа» есть поля «ObjectID», которые могут быть связаны с разными таблицами. С какой именно таблицей они связаны, указывается в поле «Table». Для создания древовидной структуры комментариев, в таблице «Комментарии» используется поле «ParentID», в котором указан комментарий-предок. Таблица «Кафедры» имеет поле «BossID», в котором указывается сотрудник с максимальными правами, а таблица «Задачи» содержит поле «EmployeeID», в котором указывается ответственный за задачу сотрудник. Также для ускорения доступа некоторые таблицы имеют поле «DeskID», в котором указана кафедра, которой принадлежит запись.

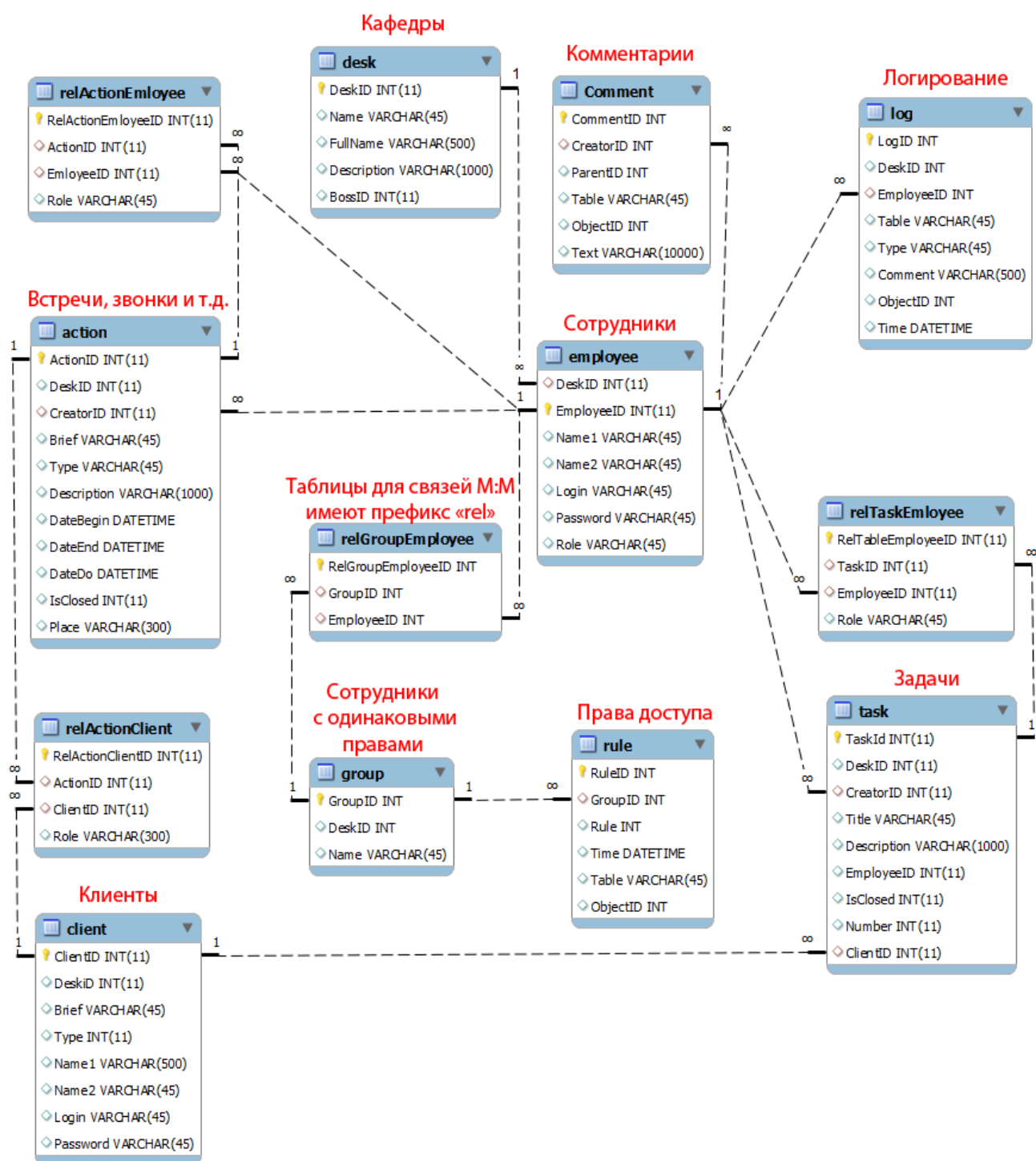


Рисунок 2. Схема модели данных

Теперь рассмотрим первый этап разработки нашей системы с точки зрения функциональных возможностей [5]. Итак, каждый пользователь должен иметь свой персонифицированный доступ в СУРП, то есть при аутентификации в системе с помощью логина и пароля, пользователь получает доступ к своей информации и создается сессия его действий. Также обеспечивается контроль доступа к данным и пользователь может просматривать только предназначенную для него информацию, ведь в системе есть информация ограниченного

использования, доступ к которой может быть разрешен ее владельцем лишь для некоторых пользователей. Возможность создавать пользователей и менять правила доступа предоставляется только администратору системы. Функциональный элемент «Задача» должен обеспечивать регистрацию задач, контроль и оповещение о задачах ответственного сотрудника. Функциональный элемент «Событие» обеспечивает похожий функционал, но с другими атрибутами, такими как время и место события. Для отчетов о проделанной работе и добавлении дополнительной информации существует функциональный элемент «Комментарий». Функциональный элемент «Кафедра» позволит структурировать данные и разбивать их по локальным кафедрам. Под словом «Взаимодействие» будем понимать создание, изменение и удаление объектов, а также выполнение выборок из набора объектов.

На первом этапе функциональность СУРП обеспечивает:

- ♣ взаимодействие с объектом «Сотрудник»;
- ♣ взаимодействие с объектом «Задача»;
- ♣ взаимодействие с объектом «Событие»;
- ♣ графический интерфейс пользователя для управления и работой над поставленными задачами и событиями;
- ♣ изменение статусов задач и добавление результатов их решения;
- ♣ возможность комментирования каждой задачи и события;
- ♣ создание отчетов по запросам данных с параметрами;
- ♣ логирование всех действий пользователя;
- ♣ настройку безопасности данных и аутентификации доступа.

После выполнения первого этапа СУРП способна ограниченно функционировать и отвечать базовым запросам пользователя. После завершения этапа и рабочего тестирования системы, проводится формирование новых требований к СУРП. Будет выполнено усовершенствование графического интерфейса и добавление возможностей работы с файлами (загрузка, прикрепление к задаче), также создание базы знаний, как отдельного источника структурированной информации. Будет развит функционал отправки сообщений, создана подсистема организации рабочего времени, а также введена система поддержки версий разработки.

На третьем этапе будет обеспечена универсальность использования системы любыми устройствами, создано приложение для планшетных компьютеров и коммуникаторов. Должен быть добавлен качественный графический дизайн центрального сайта и введена возможность использования и просмотра графических изображений в отчетах и задачах. Дополнительно будут улучшены методы поиска информации и повышена производительность системы. После каждого

этапа будет проводиться анализ возможностей и проведен опрос пользователей для развития системы в более эффективном направлении.

Сейчас система находится на стадии завершения первого этапа и прорабатываются требования следующего этапа более подробно. Проектирование системы продолжается постоянно, так как нововведения требуют новых таблиц и связей, а также изменения интерфейса. Данная работа позволяет получить практический опыт в разработке и проектировании учетной многофункциональной системы. В результате может быть получена система, наиболее полно удовлетворяющая потребностям университета в развитии научной деятельности.

Список литературы

1. Jeston J., Nelis J. Business Process Management. 2nd ed. Butterworth-Heinemann, 2008. 504 p.
2. Балдин А. В., Данчул А.Н. О реализации сервис-ориентированной архитектуры.// Инженерный вестник. МГТУ им. Н.Э. Баумана. Электрон. журн. 2012 . № 7. URL: <http://engbul.bmstu.ru/doc/457870.html>
3. Murach J., Steelman A. Murach's Java Servlets and JSP: 2nd ed. Murach & Assoc., 2008. 758 p.
4. Crawford W., Hunter J. Java Servlet Programming: 2nd ed. O'Reilly Media. 2011. 782 p.
5. Черненький В.М. Интеграция алгоритмических описаний функционирования системы. //Наука и образование. МГТУ им. Н.Э. Баумана. Электрон.журн. 2011. № 12. URL: <http://technomag.edu.ru/doc/292147.html>