

# РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ ДЛЯ АВТОМАТИЗАЦИИ ПРОЦЕССА НАЙМА ПЕРСОНАЛА

К.О. Борухина

Н.А. Шестаков

Томский политехнический университет

kob4@tpu.ru

## Введение

Для успешного роста и развития любой организации важной задачей является ее обеспечение качественным персоналом. Подбор персонала зачастую представляет собой монотонный ручной труд и для упрощения работы с кандидатами по вакансиям, необходимо наличие информационных систем, обладающих возможностью интеграции с сайтами для поиска работы и почтовыми службами, способных хранить и обрабатывать информацию по потенциальным кандидатам, и предоставлять нужные данные сотрудникам по подбору персонала.

Целью данной работы является проектирование и разработка системы для автоматизации работ по подбору персонала для ООО «РубиусГрупп».

## Бизнес-процесс работы с кандидатом

При найме на работу нового сотрудника происходят следующие процессы. Кандидаты откликаются на вакансию через сайт «hh.ru», сайт организации или через почту, отправляя свои резюме. Далее происходит первичный отсев кандидатов. Если кандидат подходит, то начинается переписка. Далее выдается тестовое задание, и происходит его проверка специалистами и в зависимости от результатов его выполнения назначается или нет техническое собеседование.

В отделе кадров много монотонной работы. Для обработки данных и работы с кандидатом используются различные приложения, что приводит к разрозненности информации. Нет единого места, где можно просмотреть и сохранить всю информацию о кандидате: переписку, результаты собеседований, личные данные. Из-за того, что вся информация хранится в разных местах, нельзя быстро находить интересующие данные.

## Описание варианта решения

Существуют различные варианты для решения поставленной задачи: данные можно обрабатывать в существующей полнофункциональной HRM-системе, другим вариантом является разработка собственных систем для хранения, обработки и предоставления данных. Второй способ можно реализовать в виде разработки настольной информационной системы или разработки веб-приложения. Был выбран последний подход ввиду ряда причин [1]:

- Для корректной работы веб-приложения не требуют установки объемного программного обеспечения на компьютеры заказчика. Необходимо лишь доступ к интернету и браузер.

- Веб-приложение не требует специальной конфигурации и администрирования.

- Обновление данных, интерфейса и функционала приложения будет происходить автоматически, нет необходимости производить синхронизацию данных.

- Работать с веб-приложением можно фактически из любой точки, где есть доступ к интернету.

## Проектирование базы данных

На основе проведенного анализа системы была спроектирована база данных, состоящая из 9 сущностей (диаграмма представлена на рис. 1). «Applicants» – таблица для хранения данных о соискателях. «Applications» – таблица, содержащая информацию о заявках. «Status» – таблица-справочник, хранящая список возможных состояний кандидата. «StatusTypes» по аналогии хранит список возможных подстатусов. «StatusHistories» предназначена для хранения архивной информации о статусах, которые данный кандидат прошел. «Vacancies» – таблица для хранения вакансий организации. «Chats» – таблица для хранения переписки. Переписка может вестись между пользователем и кандидатом и между пользователем и другим пользователем. «Questions» – таблица-справочник, хранящая список возможных вопросов, которые задаются соискателям на сайте hh.ru. «Answers» – таблица для хранения информации об ответах соискателей на вопросы на сайте hh.ru.

## Проектирование веб-приложения

При проектировании архитектуры внимание уделялось компонентной архитектуре системы. Основу системы составляют пять компонент: сервер, СУБД, сервис-поставщик данных о вакансиях и соискателях (HeadHunter API), почтовые сервисы и клиенты.

Компонент «Сервер» – сервер, на котором развёртываются следующие подкомпоненты: «DataAccessLibrary» – библиотека для доступа к данным базы данных. «HR», который представляет собой приложение, содержащее логику взаимодействия с клиентами, отправляющее им необходимую информацию, работающее непосредственно с библиотекой для доступа к данным. Данные от «HR» к клиентским компонентам будут передаваться по протоколу HTTP.

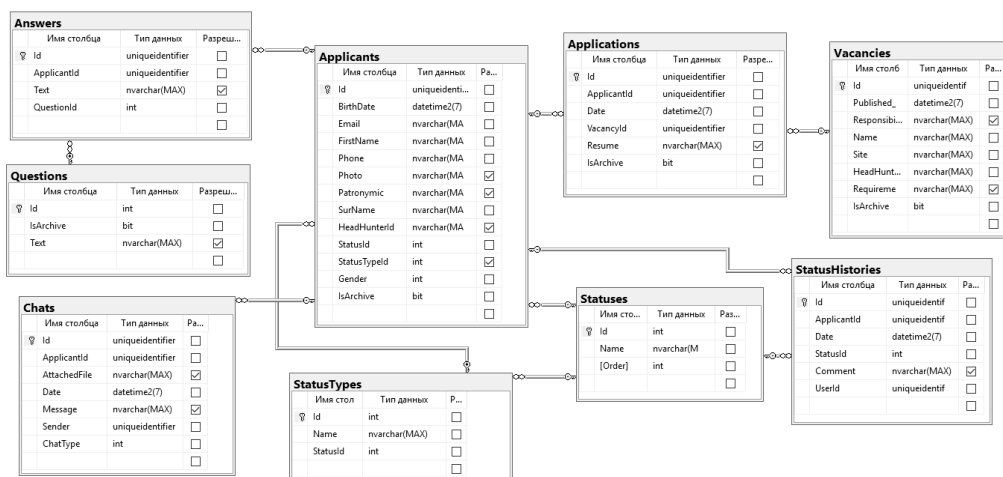


Рис. 1. Диаграмма базы данных

- Компонент «СУБД MS SQL Server 2017» - установленная на компьютере-сервере СУБД для управления базой данных.

- Компонент «ChromeExtension» - библиотека, обеспечивающая функционирование плагина для браузера Chrome. Плагин работает исключительно с библиотекой для доступа к данным.

- Компонент «MailIntegrator» - приложение, которое содержит логику для интеграции с почтой. Приложение работает с библиотекой для доступа к данным.

- В качестве клиентской компоненты будет использоваться браузер.

#### Реализация

При реализации системы использован фреймворк ASP.NET Core и шаблон MVC (Model-View-Controller) [2]. Веб-страницы генерируются на сервере при помощи фреймворка (язык разметки Razor). Также в клиентской части для интерактивности использован язык JavaScript и технология AJAX. Для создания элементов интерфейса с нестандартным поведением и внешним видом использованы такие фреймворки, как: jQuery, Telerik UI for ASP.NET Core, Bootstrap.

В качестве источника пользовательских данных в системе используется служба каталогов AD как источник, необходимый для аутентификации пользователей посредством протокола LDAP (LDAP-аутентификация).

Для связи приложения с веб-сервисами использована библиотека RestSharp, которая поддерживает автоматическую XML- и JSON-десериализацию, поддерживает вызов HTTP методов GET, POST, PUT, DELETE [3].

В компоненте «DataAccessLibrary» были разработаны 5 сервисов, для предоставления и обработки данных, связанных с пользователями

системы, кандидатами, вакансиями, статусами и чатом. Основу структуры компоненты «HR» составляют 7 контроллеров: «Account» - для реализации логики авторизации и аутентификации; «Applicants» - для управления данными, которые относятся к кандидату; «Applications» - для описания логики работы с заявками; «Statuses» - для реализации логики работы со статусами; «Chat» - контроллер, отвечающий за отображение чата на странице; «ChatApi» - apiController, отвечающий за ведение переписки пользователя с кандидатами или другими пользователями; «Vacancies» - контроллер, отвечающий за отображение списка вакансий и загрузку их с сайта «HeadHunter».

#### Заключение

Результатом выполнения работы является разработанная система для автоматизации процесса найма персонала. Система была развернута и уже используется в организации ООО «РубиусГрупп». В будущем планируется интеграция разработанной системы с сайтом организации, дополнение ее новыми функциональными возможностями.

#### Список использованных источников

1. Бейм К.О., Проектирование информационно-картографической системы гидрохимической информации // Сборник трудов XIII Международной научно-практической конференции студентов, аспирантов и молодых ученых «Молодежь и современные информационные технологии». Томск, 9-13 ноября 2015 г. – 2016 – Томск: Изд-во ТПУ. – Т. 2 – 63 - 64 с.

2. Шаблон MVC. [Электронный ресурс]. URL: <https://ru.wikipedia.org/wiki/Model-View-Controller> (дата обращения: 30.09.18).

3. RestSharp. [Электронный ресурс]. URL: <https://stackify.com/restsharp/> (дата обращения: 30.09.18).