



Столыпинский
вестник

Научная статья

Original article

УДК 004.652.4

**РАЗРАБОТКА РЕЛЯЦИОННОЙ БАЗЫ ДАННЫХ ДЛЯ СИСТЕМЫ
АВТОМАТИЗАЦИИ ТЕХНИЧЕСКОЙ ПОДДЕРЖКИ ПОЛЬЗОВАТЕЛЕЙ
DEVELOPMENT OF A RELATIONAL DATABASE FOR AUTOMATION
SYSTEM OF TECHNICAL SUPPORT FOR USERS**

Хмылев Кирилл Валерьевич, студент 2 курса магистратуры, факультет "Информатика и системы управления", МГТУ им. Н.Э. Баумана, Россия, г. Москва

Научный руководитель: Сотников Алексей Александрович

Khmylev Kirill Valerievich, 2nd year master's student, Faculty of Informatics and Control Systems, Moscow State Technical University. N.E. Bauman, Russia, Moscow
Scientific adviser: Sotnikov Alexey Alexandrovich

Аннотация. Данная статья посвящена разработке реляционной базы данных для системы автоматизации технической поддержки. В рамках работы была проанализирована предметная область и выделены основные сущности, обладающие определенным набором свойств в рамках предметной области. В результате была разработана реляционная база данных для системы обработки сервисных событий с помощью РСУБД MS SQL Server 2019.

Abstract. This article is devoted to the development of a relational database for a technical support automation system. As part of the work, the subject area was

analyzed and the main entities that have a certain set of properties within the subject area were identified. As a result, a relational database was developed for the service event processing system using RDBMS MS SQL Server 2019.

Ключевые слова: *системы автоматизации, техническая поддержка пользователей, проектирование, разработка базы данных, реляционная база данных.*

Keywords: *automation systems, user technical support, design, database development, relational database.*

Введение

В рамках работы было выделено 10 сущностей в ИС: Обращение(инцидент), действие по инциденту, пользователь ИС, группа пользователей, оператор ИС, группа операторов, статус, приоритет, тип действия инцидента, услуга. Каждая сущность независима, но информация передается между всеми. Пользователь ИС заводит обращение в ИС по выбранной им услуге. Назначенная группа операторов обрабатывает полученное обращение, выполняя определённые действия и формируя ответ на него. Пользователь получает ответ на созданное им обращение.

Продуманная структура и реализация информационной системы помогает добиться следующих целей:

- улучшить взаимодействие между службой поддержки и пользователями(клиентами);
- сократить время обслуживания инцидента;
- усовершенствовать процедуры отслеживания и отработки инцидентов;
- устранить ошибки, связанные с человеческим фактором;
- автоматизировать рутинные процессы;
- получать детальную статистику работы всего подразделения и каждого сотрудника.

Рисунок 1. Схема базы данных

Сущность **incident** состоит из полей:

- Incident_id – первичный ключ, уникальный id инцидента;
- Inc_status_id – статус инцидента, внешний ключ к таблице status;
- Date_logged – дата регистрации инцидента;
- Usr_id – id пользователя ИС, внешний ключ к таблице usr;
- Usr_serv_dept_id – id группы пользователя ИС, внешний ключ к таблице user_dept;
- Ass_opr_usr_id – id назначенного оператора на инцидент, внешний ключ к таблице opr_usr;
- Ass_svd_id – id группы оператора, назначенного на инцидент, внешний ключ к таблице serv_dept;
- Inc_priority_id – id приоритета инцидента, внешний ключ к таблице priority;
- Inc_service_id – id услуги, выбранной пользователем ИС, внешний ключ к таблице service;
- Inc_resolve_opr_usr_id – id оператора, решившего инцидент, внешний ключ к таблице opr_usr;
- Inc_resolve_serv_dept_id – id группы оператора, решившего инцидент, внешний ключ к таблице serv_dept;
- Inc_raiting – оценка решения инцидента, выставляемая пользователем (ограничение на значение оценки – диапазон 1 – 5);
- Inc_remark – описание инцидента;
- Event_type – тип обращения.

Сущность **act_reg** состоит из полей:

- Act_reg_id – id действия, первичный ключ к таблице act_reg;
- Act_type_id – тип действия, внешний ключ к таблице act_type;
- Incident_id – id инцидента, внешний ключ к таблице incident;
- Serv_dept_id – id группы оператора, выполнившего действие, внешний ключ к таблице serv_dept;

- Opr_usr_id – id оператора, выполнившего действие, внешний ключ к таблице opr_usr;
- Usr_id – id пользователя, выполнившего действие, внешний ключ к таблице usr;
- Ass_opr_usr_id – id оператора, назначенного на инцидент, внешний ключ к таблице opr_usr;
- Ass_svd_id – id группы оператора, назначенного на инцидент, внешний ключ к таблице serv_dept;
- Date_logged – дата регистрации действия;
- Act_remark – описание действия;
- Event_type – тип инцидента в котором было выполнено действие.

Сущность **opr_usr** состоит из полей:

- Opr_usr_id – id оператора, первичный ключ к таблице opr_usr;
- Opr_usr_login – логин УЗ оператора в ИС;
- Opr_usr_password – пароль УЗ оператора в ИС;
- Opr_usr_name – имя оператора;
- Opr_usr_surname – фамилия оператора;
- Opr_usr_email – email адрес оператора;
- Opr_usr_tele – номер телефона оператора;
- Opr_usr_registration_date – дата регистрации УЗ оператора;
- Opr_usr_serv_dept_id – группа оператора, внешний ключ к таблице serv_dept.

Сущность **usr** состоит из полей:

- Usr_id – id пользователя, первичный ключ к таблице usr;
- usr_login – логин УЗ пользователя в ИС;
- usr_password – пароль УЗ пользователя в ИС;
- usr_name – имя пользователя;
- usr_surname – фамилия пользователя;
- usr_email – email адрес пользователя;

- `usr_tele` – номер телефона пользователя;
- `usr_registration_date` – дата регистрации УЗ пользователя;
- `usr_serv_dept_id` – группа пользователя, внешний ключ к таблице `user_dept`.

Сущность **user_dept** состоит из полей:

- `user_dept_id` – id группы пользователя, первичный ключ к таблице `user_dept`;
- `user_dept_name` – название группы пользователя.

Сущность **serv_dept** состоит из полей:

- `serv_dept_id` – id группы оператора, первичный ключ к таблице `serv_dept`;
- `serv_dept_name` – название группы оператора.

Сущность **act_type** состоит из полей:

- `act_type_id` – id тип действия, первичный ключ к таблице `act_type`;
- `act_type_n` – название типа действия.

Сущность **priority** состоит из полей:

- `priority_id` - id приоритета действия, первичный ключ к таблице `priority`;
- `priority_name` – название приоритета действия.

Сущность **status** состоит из полей:

- `status_id` - id статуса действия, первичный ключ к таблице `status`;
- `status_name` – название статуса действия.

Сущность **service** состоит из полей:

- `service_id` - id услуги, первичный ключ к таблице `service`;
- `service_name` – название услуги;
- `ass_svd_id` – id группы операторов, автоматически назначаемой при регистрации инцидента.

Также в базе данных были выделены роли: роль пользователя, роль оператора.

Роль пользователя ИС стандартная:

- просмотр полей таблицы `act_reg`;
- просмотр и вставка полей в таблице `incident`.

Роль оператора зависит от его группы. Для примера были созданы две роли:

- Отдел бухгалтерии;
- Отдел канцелярии.

Отделу бухгалтерии разрешено:

- Просмотр, вставка, обновление полей в таблице act_reg;
- Просмотр, вставка, обновление полей в таблице incident.

Отделу канцелярии разрешено:

- Просмотр, обновление полей в таблице act_reg;
- Просмотр, обновление полей в таблице incident.

Данные роли были созданы в БД с помощью среды управления БД - SQL Server Management Studio.

Разработка БД

Основные функции в ИС – действия над инцидентами, которые осуществляют пользователи и операторы. С помощью триггеров к таблицам act_reg и incident была реализована логика работы действий.

У каждого пользователя и оператора есть доступные действия для операций над инцидентом. Для пользователя:

- Действие «Оценить»;
- Действие «Предоставить дополнительную информацию»;
- Действие «Уточнить статус заявки».

Для оператора:

- Действие «Взять в работу»;
- Действие «Решить»;
- Действие «Открыть повторно»;
- Действие «Запрос дополнительной информации»;
- Действие «Отклонить»;
- Действие «Назначить».

Основные функции триггера act_reg_INSERT_UPDATE:

- Определение группы оператора по его id;
- Реализация действия «Взять в работу»;
- Реализация действия «Решить»;
- Реализация системного действия «Закреть»;
- Реализация действия «Отклонить»;
- Реализация действия «Открыть повторно»;
- Реализация действия «Оценить».

Основные функции триггера incident_INSERT_UPDATE:

- Автоматическое назначение ответственной группы-исполнителя по выбранному сервису пользователем;
- Автоматическое определение группы пользователя по его id.

Также были реализованы процедуры для регистрации инцидентов пользователем Incidents_registration() и процедура для запуска действий action().

Для группировки информации в виде отчётов удобно использовать представления БД. Для примера было разработано представление Rejected_incidents_last_week – отчёт отклонённых инцидентов за последнюю неделю. На рисунке 2 показан пример отчёта.

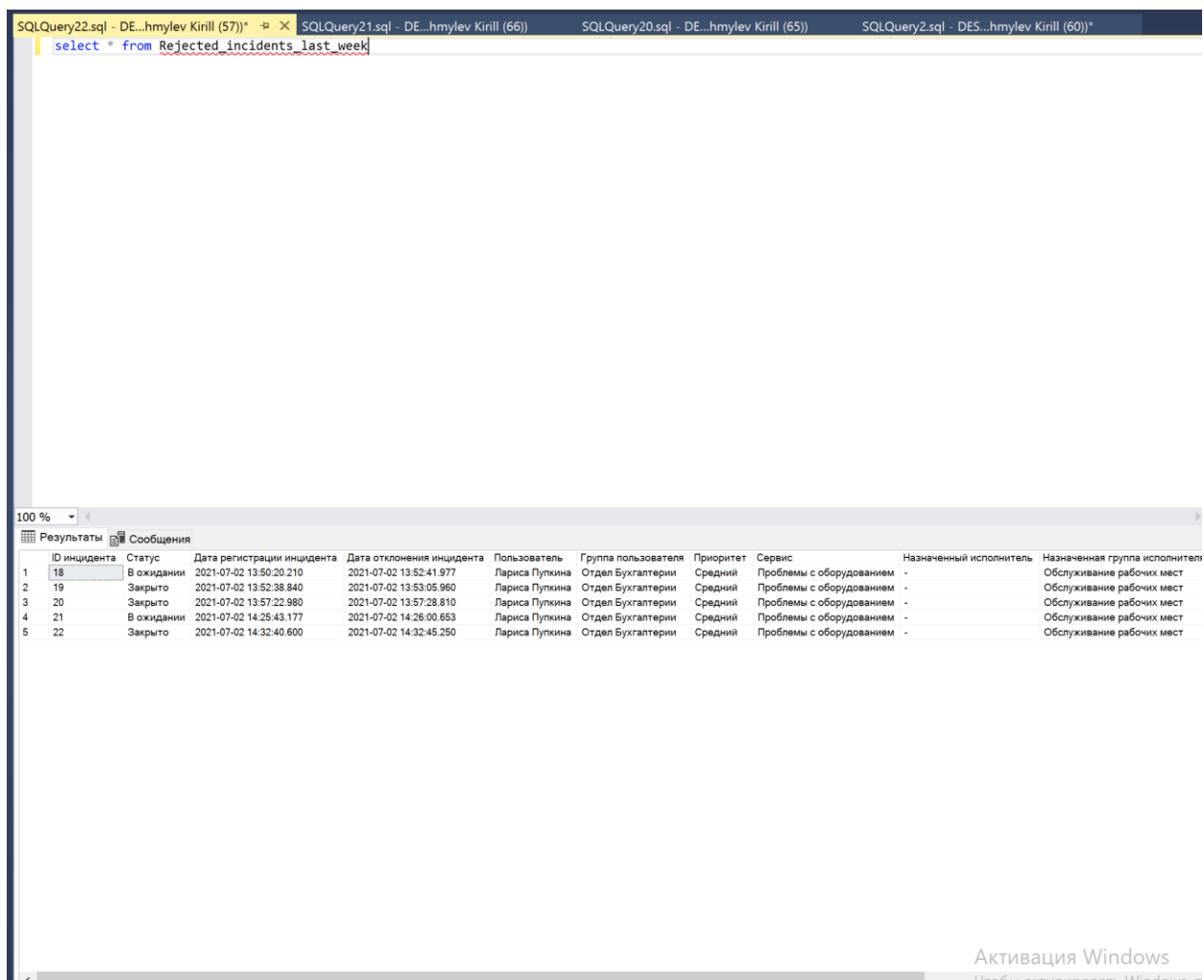


Рисунок 2. Пример отчёта, сформированного при выполнении представления `Rejected_incidents_last_week`

В результате работы был разработан прототип реляционной базы данных для системы обработки сервисных событий с помощью PCУБД MS SQL Server 2019.

Литература

1. Кляйн, Д. SQL. Справочник / Д. Кляйн
2. Петкович, Д. Microsoft SQL Server 2012. Руководство для начинающих
3. Лукин, В.Н. Введение в проектирование баз данных
4. Пирогов, В. Информационные системы и базы данных: организация и проектирование: Учебное пособие
5. Мейер М. Теория реляционных баз данных
6. Грабер, Мартин SQL для простых смертных / Мартин Грабер

References

1. Klein, D. SQL. Directory / D. Klein
2. Petkovic, D. Microsoft SQL Server 2012 Beginner's Guide
3. Lukin, V.N. Introduction to Database Design
4. Pirogov, V. Information systems and databases: organization and design: Textbook
5. Meyer M. Theory of relational databases
6. Graber, Martin SQL for mere mortals / Martin Graber

© Хмылев К.В., 2023 Научный сетевой журнал «Столыпинский вестник» 4/2023

Для цитирования: Хмылев К.В. РАЗРАБОТКА РЕЛЯЦИОННОЙ БАЗЫ ДАННЫХ ДЛЯ СИСТЕМЫ АВТОМАТИЗАЦИИ ТЕХНИЧЕСКОЙ ПОДДЕРЖКИ ПОЛЬЗОВАТЕЛЕЙ// Научный сетевой журнал «Столыпинский вестник» 4/2023