МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ  
Учреждение образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ   
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет Информационных технологий

Кафедра Информационных систем и технологий

Специальность 1-40 05 01 «Информационные системы и технологии»

**ОТЧЁТ**

**по дисциплине** "Методы сбора, хранения, обработки и анализа данных"

**Лабораторная работа №1**

**Проектирование базы данных.**

Студент: Стрелковская В. А.

ФИТ 3 курс 2 группа

Преподаватель: Комкова А. В.

**1.Список типичных бизнес-задач:**

* Учет лицензий и их стоимости: отслеживание данных о приобретённых лицензиях на программное обеспечение, включая стоимость каждой лицензии.
* Мониторинг срока действия лицензий: оперативный контроль за датой истечения срока действия каждой лицензии для предотвращения использования просроченных лицензий.
* Анализ использования ПО: оценка популярности и частоты использования различных типов программного обеспечения для принятия решений по оптимизации лицензионных расходов.
* Управление вендорами/поставщиками ПО: содержание данных о вендорах/поставщиках программного обеспечения для сотрудничества и обеспечения актуальности лицензий.
* Расчет общей стоимости лицензий в разрезе времени: формирование отчетов о затратах на лицензии за определенные периоды (месяц, квартал, полгода, год).

**2.Список пользователей**

* Администраторы: осуществляют администрирование системы управления лицензиями, имеют доступ к общей базе данных.
* Менеджеры по закупкам: занимаются планированием и закупкой лицензий, отслеживают бюджет и статус существующих лицензий.
* IT-специалисты: отвечают за установку и обновление программного обеспечения, могут запрашивать информацию о доступных лицензиях.
* Пользователи: зависит от специфики организации, но это могут быть конечные пользователи, использующие программное обеспечение, для которого управляются лицензии.

**3.Оценка сроков проекта:**

1. Подготовительный этап (2-4 недели):
   * определение требований: Сбор и анализ функциональных возможностей проекта;
   * проектирование: Разработка структуры базы данных и архитектуры системы;
   * формирование команды: Набор и обучение разработчиков и дизайнеров.
2. Разработка (12-22 недель):
   * разработка БД (4-8 недель): настройка базы данных, создание таблиц и индексов в соответствии с проектной моделью, разработка хранимых процедур, триггеров и функций (при необходимости);
   * разработка приложения (6-10 недель): создание пользовательских интерфейсов для администратора, владельцев лицензий и конечных пользователей, создание системы регистрации и аутентификации пользователей, реализация функциональности управления пользователями, лицензиями и активацией лицензий.
   * интеграция с приложением (2-4 недели): связывание базы данных с приложением (бэкенд), разработка API для взаимодействия с базой данных, тестирование функциональности связи приложения с базой данных.
3. Тестирование и отладка (4-6 недель):
   * тестирование всех компонентов системы на предмет ошибок и неполадок;
   * отладка и исправление обнаруженных проблем;
   * тестирование безопасности и защиты данных.
4. Запуск и внедрение (1-2 недели):
   * Переход на рабочую версию платформы;
   * мониторинг процесса запуска и решение возникающих проблем.
5. Сопровождение и развитие (постоянно):
   * постоянное обновление и сопровождение системы;
   * разработка новых функциональных возможностей и улучшений в ответ на потребности пользователей.

Общая продолжительность проекта может колебаться в зависимости от сложности и объема функциональности, но предполагаемый срок выполнения проекта составляет примерно 20-35 недель. Стоит учесть, что это всего лишь оценка, и фактические сроки могут изменяться в зависимости от ряда факторов, таких как доступность ресурсов и изменения требований проекта.

**4.Требуемые ресурсы:**

1. Разработчики:
   * фронтенд-разработчики для создания пользовательских интерфейсов;
   * бэкенд-разработчики для разработки серверной части приложения и интеграции с базой данных.
2. Дизайнеры:
   * специалисты по дизайну интерфейсов для создания пользовательского опыта и интерфейсов приложения.
3. Бизнес-аналитики:
   * аналитики, помогающие определить требования заказчика, проводя анализ рынка и потребительских привычек, чтобы обеспечить разработку приложения, соответствующего потребностям рынка.
4. Тестировщики:
   * специалисты по тестированию для проверки приложения на ошибки, обеспечивая его надежность, безопасность и корректную работу. Они также проводят тестирование производительности и совместимости.
5. Оборудование и хостинг:
   * серверное оборудование или хостинг-провайдер для размещения базы данных и приложения;
   * компьютеры и рабочие места для разработчиков, тестировщиков и аналитиков.
6. Программное обеспечение:
   * система управления БД
   * среда разработки
   * инструменты для тестирования
   * платформа для совместной работы
7. Бюджет:
   * финансовые ресурсы для оплаты заработной платы сотрудникам;
   * закупка необходимых лицензий и инструментов разработки;
   * расходы на хостинг и оборудование;
   * различные операционные расходы, включая маркетинг и рекламу (если требуется).

Эти ресурсы необходимы для успешной разработки, тестирования и внедрения системы управления лицензиями на ПО. Важно также учесть, что бюджет должен быть выделен на всю продолжительность проекта, включая сопровождение и развитие.

**5.Обеспечение безопасности:**

1. Регистрация и аутентификация пользователей:
   * использование шифрования при передаче данных, таких как пароли;
   * поддержка двухфакторной аутентификации для повышения безопасности;
   * ограничение числа неудачных попыток входа и блокировка аккаунта при нескольких неудачных попытках;
   * регулярное обновление паролей и требование к их сложности.
2. Управление базой лицензий:
   * авторизация доступа к данным о лицензиях с использованием ролевой модели доступа;
   * шифрование лицензионных ключей и других конфиденциальных данных;
   * отслеживание доступа к данным для выявления несанкционированных действий;
   * резервное копирование и защита данных для предотвращения потери информации.
3. Управление продуктами ПО:
   * контроль доступа к административным функциям для владельцев лицензий;
   * проверка изменений данных о продуктах ПО;
   * шифрование ценовой информации и других конфиденциальных данных;
   * защита от SQL-инъекций и других веб-уязвимостей.
4. Мониторинг и анализ:
   * анономность данных пользователей для обеспечения конфиденциальности;
   * использование мониторинга безопасности для выявления подозрительной активности;
   * регулярное обновление системы для устранения известных уязвимостей.
5. Решение технических проблем и вопросов пользователей:
   * аутентификация пользователей при обращении в службу поддержки;
   * шифрование сообщений и данных, обмениваемых между пользователями и службой поддержки;
   * система журналирования событий для регистрации запросов и решения проблем.

**6.UML-диаграмма**

Данная диаграмма демонстрирует возможности пользователей.

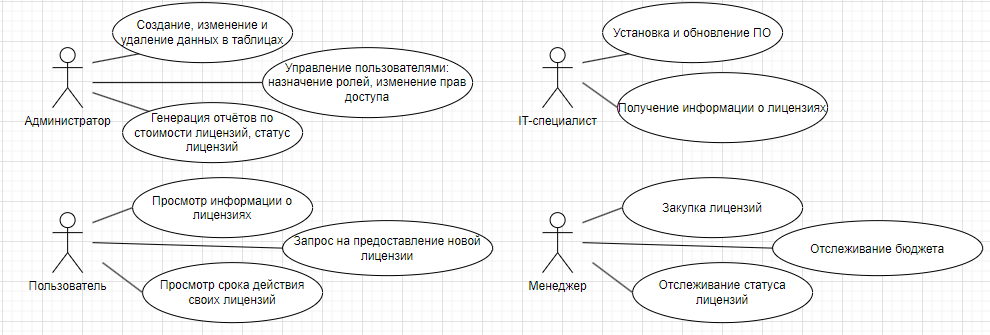


Рисунок 5.2 – UML-диаграмма с точки зрения использования

Администратор, пользователь системы имеют следующие права:

1. Администратор:
   1. создание, редактирование и управление лицензиями;
   2. управление пользователями;
   3. выполнение других административных задач.
2. Пользователь:
   1. просмотр доступных лицензий в системе;
   2. покупка и аренда лицензий на программное обеспечение;
   3. просмотр своих лицензий.
3. IT-специалист:
   1. установка и обновление программного обеспечения
   2. просмотр информации о лицензиях
4. Менеджер по закупкам:
   1. Закупка лицензий
   2. Отслеживание бюджета
   3. Отслеживание статуса лицензий

Эти роли лучше отражают функциональные возможности и права в контексте системы управления лицензиями на программное обеспечение. Теперь вы можете использовать эту информацию для создания UML-диаграммы с соответствующими ролями и их правами.

**6.Физическая схема**

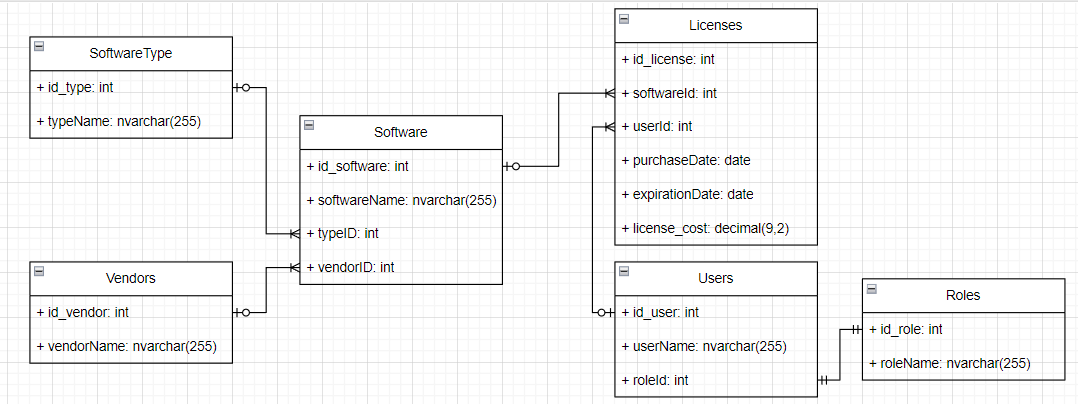


Рисунок 6.1– физическая схема MS SQL

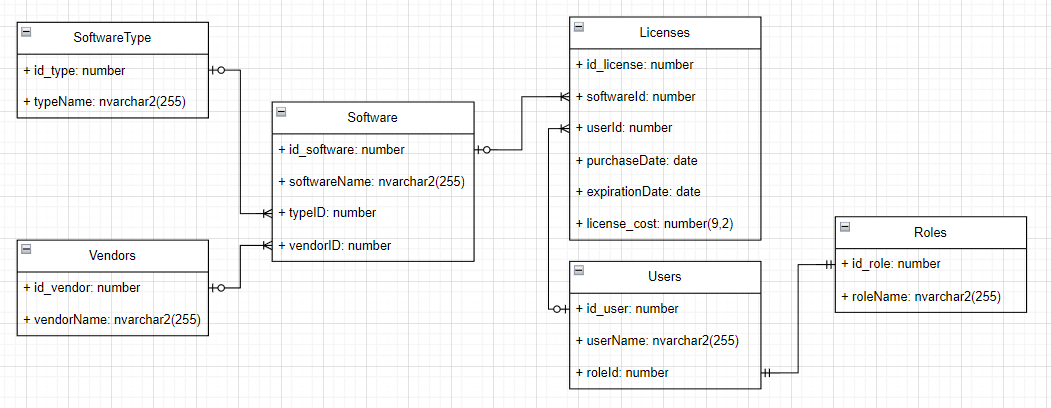


Рисунок 6.2– физическая схема ORACLE

**7.Логическая схема**

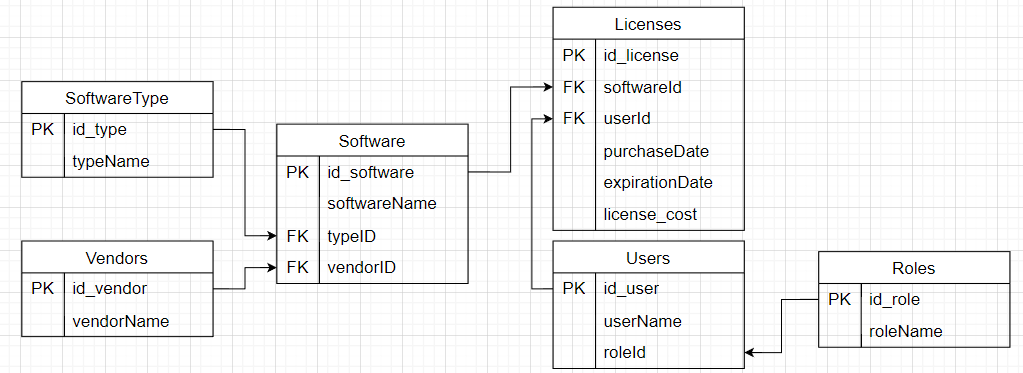
****

Рисунок 7.1– Логическая схема

**Нормальные формы:**

* **1 НФ**: Все атрибуты имеют атомарные значения.
* **2 НФ**: Все неключевые атрибуты полностью функционально зависят от первичного ключа.
* **3 НФ**: Нет транзитивных зависимостей между неключевыми атрибутами.
* **BCNF**: Для каждого функционального зависимости X → Y, X должно быть суперключом.
* **4 НФ**: Отсутствие многозначных зависимостей.

**Денормализованные атрибуты:**

* Оценить необходимость денормализации для повышения производительности.
* Возможно, стоит сохранить часто используемые данные в одной таблице для ускорения доступа.

**Обеспечение безопасности:**

* Определить роли и права доступа для разных пользователей.
* Использовать шифрование для хранения конфиденциальных данных.