

СОДЕРЖАНИЕ

ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ, СОКРАЩЕНИЯ	4
ВВЕДЕНИЕ.....	5
1 ФУНКЦИОНАЛЬНАЯ ДИАГРАММА IDEF0	6
1.1 ФУНКЦИОНАЛЬНАЯ МОДЕЛЬ С ТОЧКИ ЗРЕНИЯ РЕСУРС-МЕНЕДЖЕРА.....	6
1.2 ФУНКЦИОНАЛЬНАЯ МОДЕЛЬ С ТОЧКИ ЗРЕНИЯ КАНДИДАТА	13
2 ПРОЕКТИРОВАНИЕ СТРУКТУРЫ БАЗЫ ДАННЫХ.....	15
2.1 КОНЦЕПТУАЛЬНОЕ ПРОЕКТИРОВАНИЕ	16
2.2 ЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ	17
2.3 ФИЗИЧЕСКОЕ ПРОЕКТИРОВАНИЕ	19
ЗАКЛЮЧЕНИЕ	21
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	22

ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ, СОКРАЩЕНИЯ

SQL (Structured Query Language) - декларативный язык программирования, применяемый для создания, модификации и управления данными в реляционной базе данных, управляемой соответствующей системой управления базами данных.

IDE (Integrated Development Environment) — система программных средств, используемая программистами для разработки программного обеспечения.

ВВЕДЕНИЕ

Система контроля языка запросов к базе данных (СКЗБД) требует проектирования достаточно сложной базы данных, в связи с тем, что система реализует обширный функционал и предполагает дальнейшую масштабируемость и дальнейшую интеграцию с другими системами. Кроме того, разрабатываемая система предполагает использование различных баз данных для проведения тестирования кандидатов, в связи с чем система будет требовать несколько баз данных для работы с таким количеством информации.

Для проектирования базы данных, которая будет использоваться непосредственно приложением, требуется досконально разобраться в функциональной модели работы системы. Для данной цели лучше всего подойдет нотация IDEF0, так как она позволит в графическом формате изобразить все бизнес-процессы системы для последующего анализа.

1 Функциональная диаграмма IDEF0

Структурно-функциональная модель системы на основе методики IDEF0 позволит досконально разобраться в функциональных характеристиках разрабатываемой системы и рассмотреть все бизнес-процессы системы. Рассмотрим СКЗБД с нескольких точек зрения: с точки зрения ресурс-менеджера компании, который имеет потребность в оценке знаний кандидатов, и с точки зрения кандидата, который проходит тестирование в системе.

1.1 Функциональная модель с точки зрения ресурс-менеджера

В соответствии с требованиями, предъявляемыми к разрабатываемой системе, можно выделить несколько входных данных:

- информация о кандидате – информация о навыках кандидата полученная из резюме или общения с сотрудником отдела кадров.
- Пул существующих вопросов – список вопросов, которые были созданы прежде и могут быть использованы для комплекта заданий.
- Пул существующих объектов БД – информация об объектах, которые существуют в базе данных и могут быть использованы для новых заданий.
- Практический опыт – опыт ресурс-менеджера, полученный в ходе повседневной работы.

В качестве управляющих данных можно выделить информацию о навыках, необходимых для вакансии, так как данная информация играет решающую роль в выборе кандидата.

Механизмами являются две основные системы:

- Тестирующая система – разрабатываемая система.
 - СУБД – системы управления базами данных, которые будут использоваться для проверки правильности выполнения задания.
- Определяются в соответствии с техническим заданием.

Выходными данными являются результаты тестирования, которые будут использованы для принятия решения о соответствии кандидата на должность.

На основе представленных данных может быть разработана начальная контекстная диаграмма (рисунок 1).

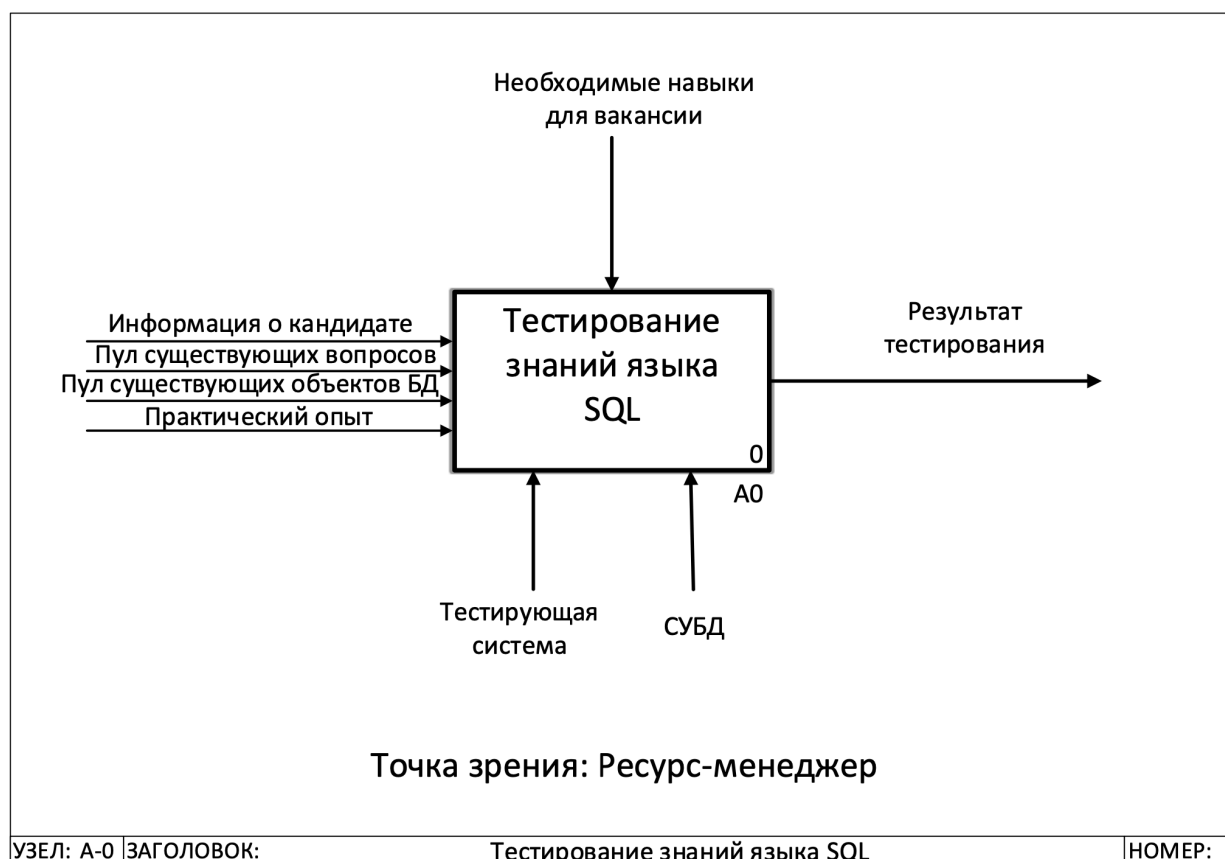


Рисунок 1 – Начальная контекстная диаграмма A-0

При детализации диаграммы A-0 можно выделить несколько основных функциональных блоков:

1. Просмотр существующих заданий – получение списка уже существующих заданий и информации о них.
2. Создание нового задания – создание объектов, написание задания и эталонного решения данного задания.
3. Создание комплекта заданий – сборка нескольких заданий в комплект для отправки кандидату.

4. Отправка комплекта заданий кандидату
5. Проведение тестирования
6. Получение результатов – проверка полученного решения кандидата и сравнение с эталонным решением.

Диаграмма A0 представлена на рисунке 2.

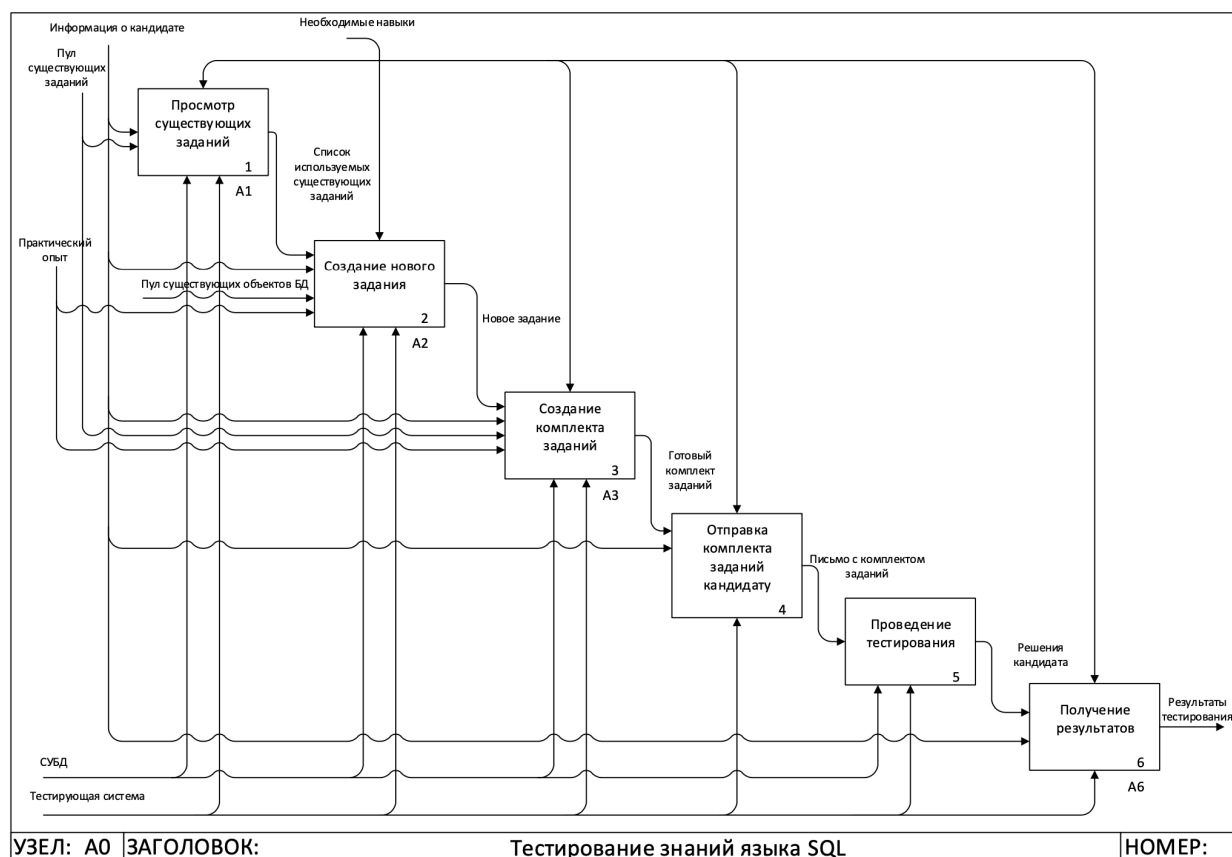


Рисунок 2 – Диаграмма A0. Тестирование знаний языка SQL

Функциональный блок «Просмотр существующих заданий» может быть детализирован в следующие функциональные блоки:

1. Просмотр ранее созданных заданий
2. Просмотр заданий других менеджеров с открытым доступом
3. Принятие решения об использовании уже созданных заданий

Данная детализация представлена на рисунке 3.

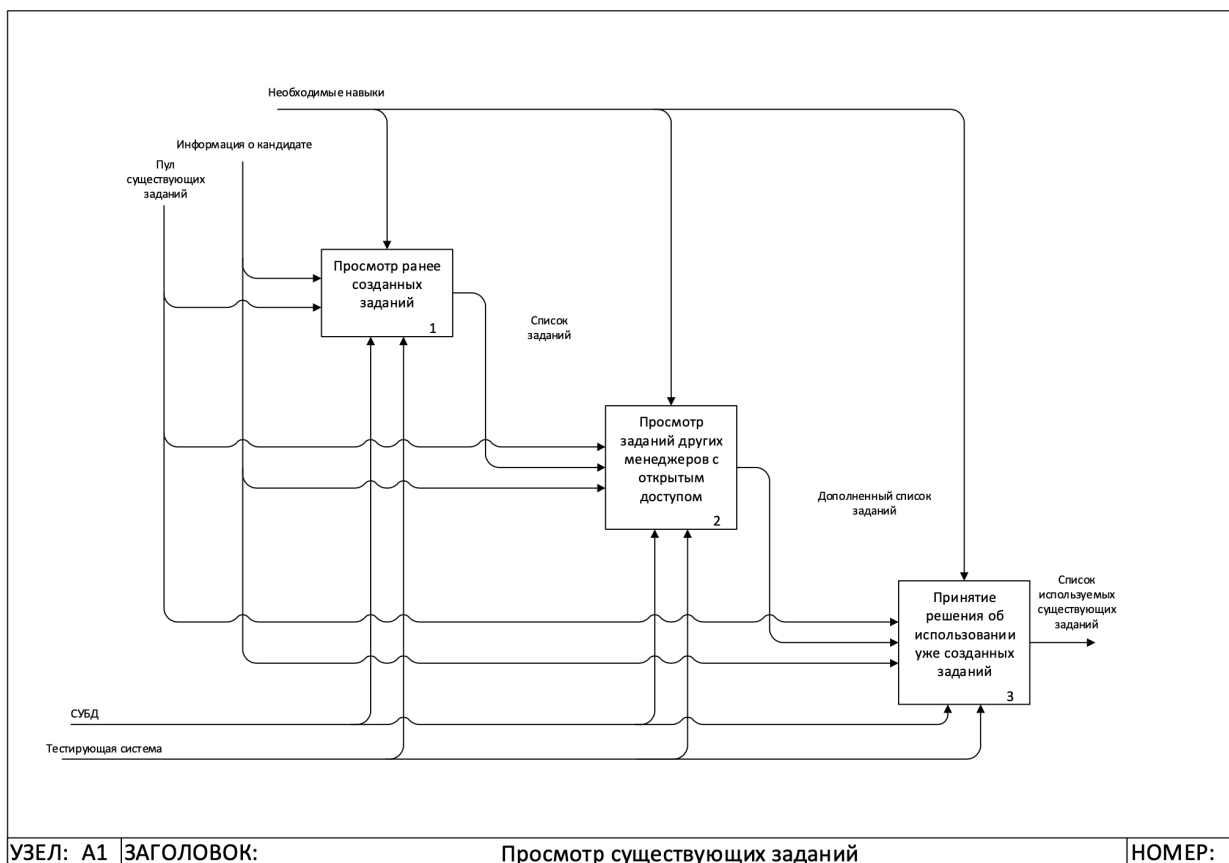


Рисунок 3 – Диаграмма A1. Просмотр существующих заданий

Самым сложным функциональным блоком системы является блок о создании новых заданий. Данный блок состоит из следующих шагов:

1. Выбор схемы БД или создание новой – может быть выбрана схема с объектами, которая уже использовалась для создания заданий или может быть создана совершенно новая схема без объектов. В результате данного шага будут получены логин с паролем для прямого доступа к базе данных для точечной модификации объектов и данных.
2. Создание объектов – добавление объектов в базу данных для последующего составления заданий на их основе. Может быть сделано посредством загрузки sql файла в интерфейсе тестирующей системы или с помощью прямого подключения к базе данных через программное обеспечение соответствующих СУБД. Данный функциональный блок будет дополнительно детализирован далее.

3. Написание условия задания – написание задания для кандидата в текстовом формате с возможностью добавления изображений (например, схемы СУБД).
4. Написание эталонного запроса – написание эталонного решения на написанное прежде задание. Данное решение будет использоваться для проверки правильности решения кандидата путем запуска решений и сверки результатов выполнения.
5. Задание времени на выполнение задания – установка максимального времени решения задания.
6. Проставление меток на задание – добавление меток о проверяемых знаниях кандидата и сложности к заданию.
7. Сохранение задания.

Диаграмма A2 представлена на рисунке 4.

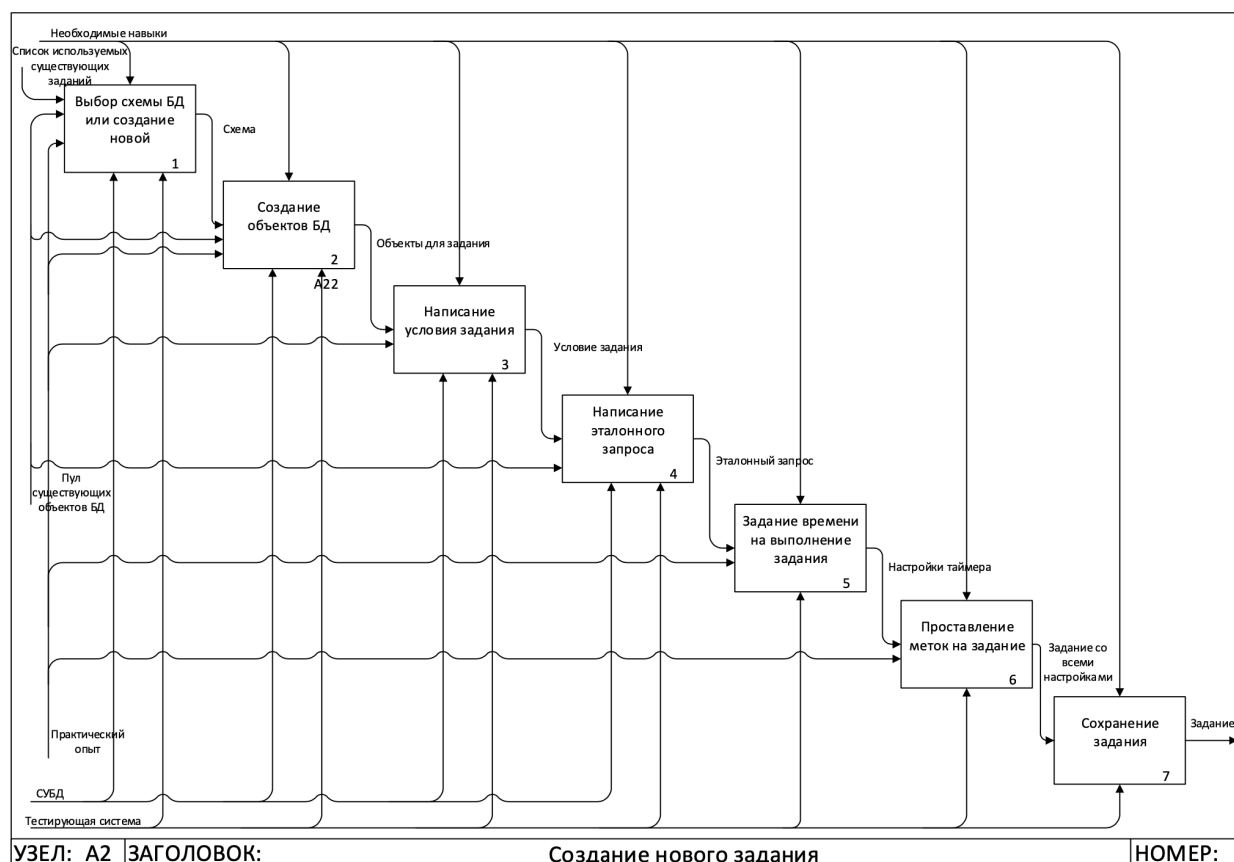


Рисунок 4 – Диаграмма A2. Создание нового задания.

Как было упомянуто ранее, блок «Создание объектов БД» требует дополнительной детализации. Данный функциональный блок состоит из:

1. Получение логина и пароля для подключения к БД.
2. Анализ существующих объектов.
3. Добавление объектов с помощью sql файла – загрузка и выполнение sql файла через разрабатываемую систему.
4. Точечное исправление объектов и данных в БД – исправление объектов БД с помощью прямого подключения к базе данных через программное обеспечение соответствующих СУБД.

Данная детализация представлена на рисунке 5.

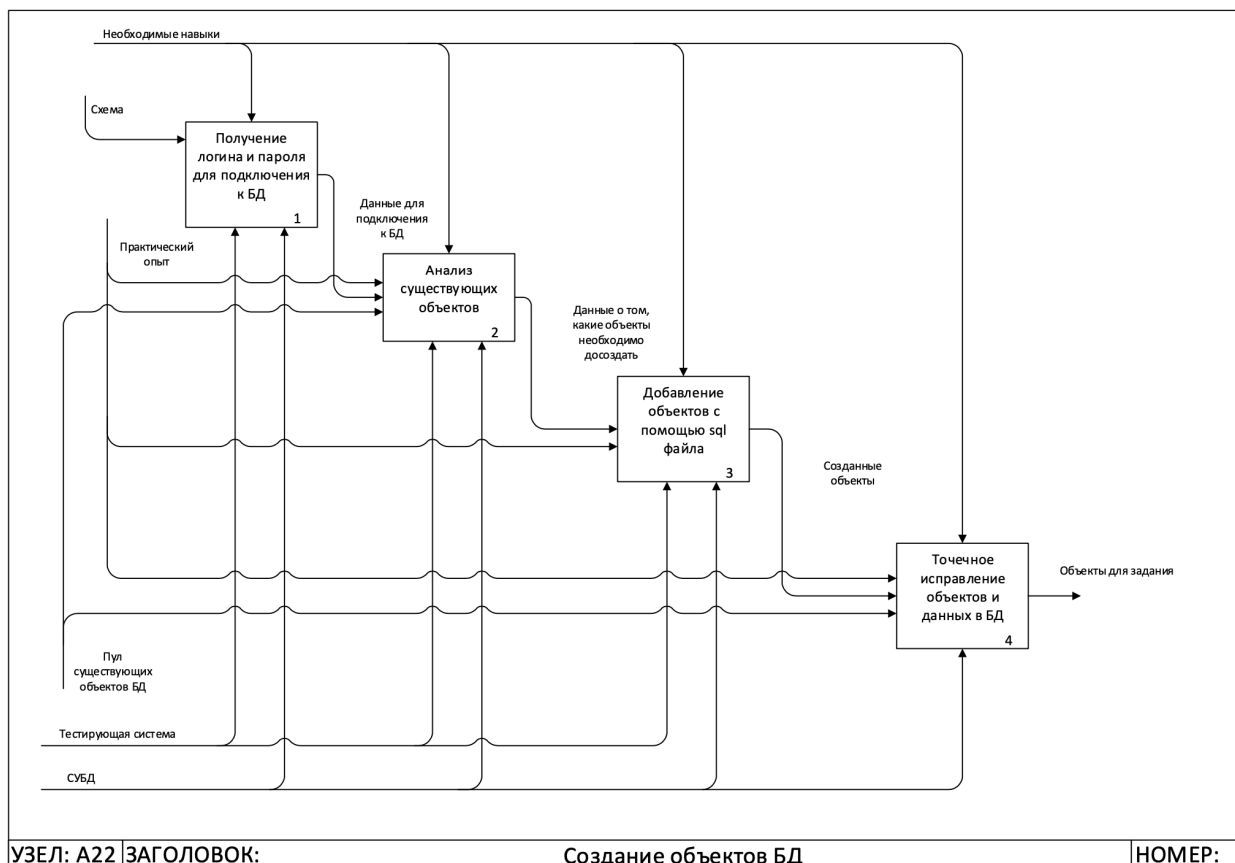


Рисунок 5 – Диаграмма A22. Сохранение объектов в БД

Блок «Создание комплекта заданий» (рисунок 2) может быть детализирован до следующих блоков:

1. Выбор заданий для комплекта.
2. Расположение их по порядку.
3. Сохранение комплекта.

Описанная детализация представлена на рисунке 6.

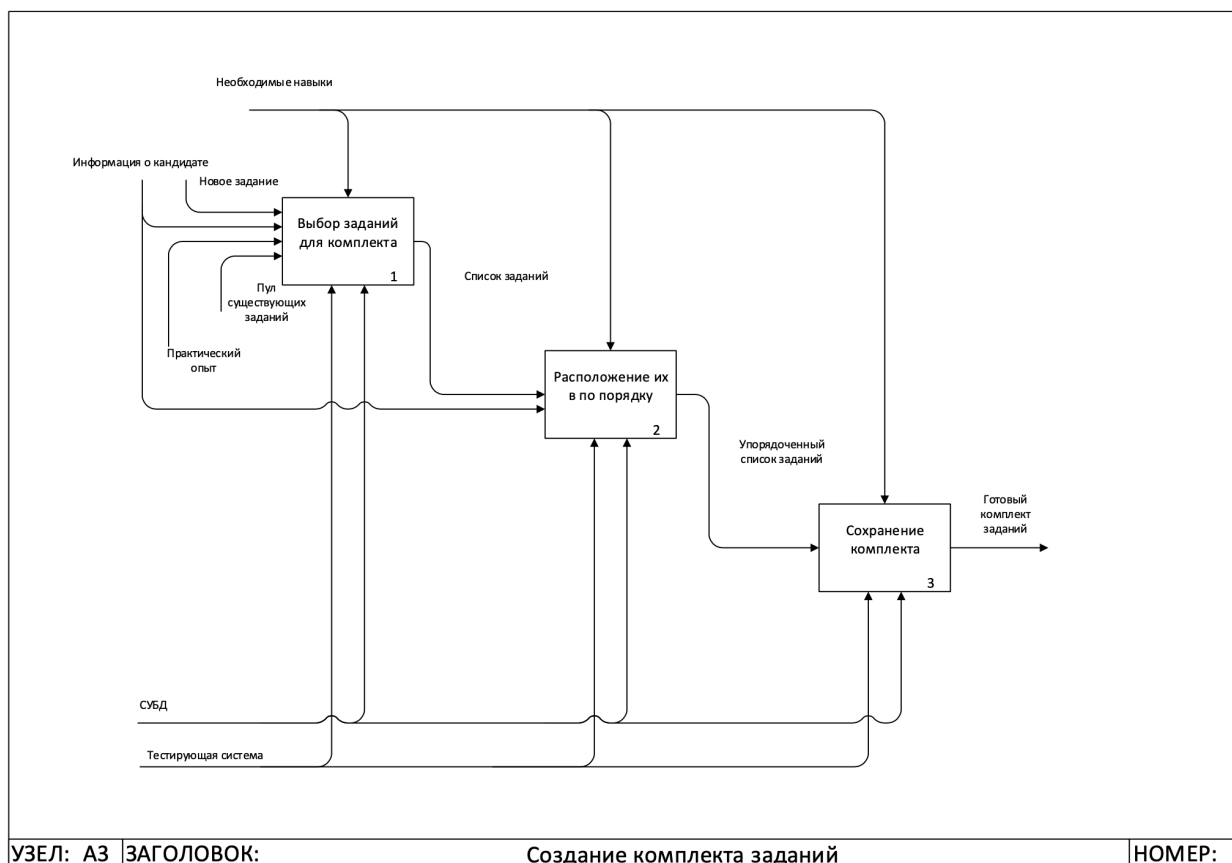


Рисунок 6 – Диаграмма А3. Создание комплекта заданий

Последним блоком, нуждающимся в детализации, является блок «Получение результатов» (рисунок 2). Блоки детализации представлены далее:

1. Сверка решений кандидата с эталонными запросами.
2. Сбор статистики.
3. Показ результатов кандидатов.

Детализация представлена на рисунке 7.

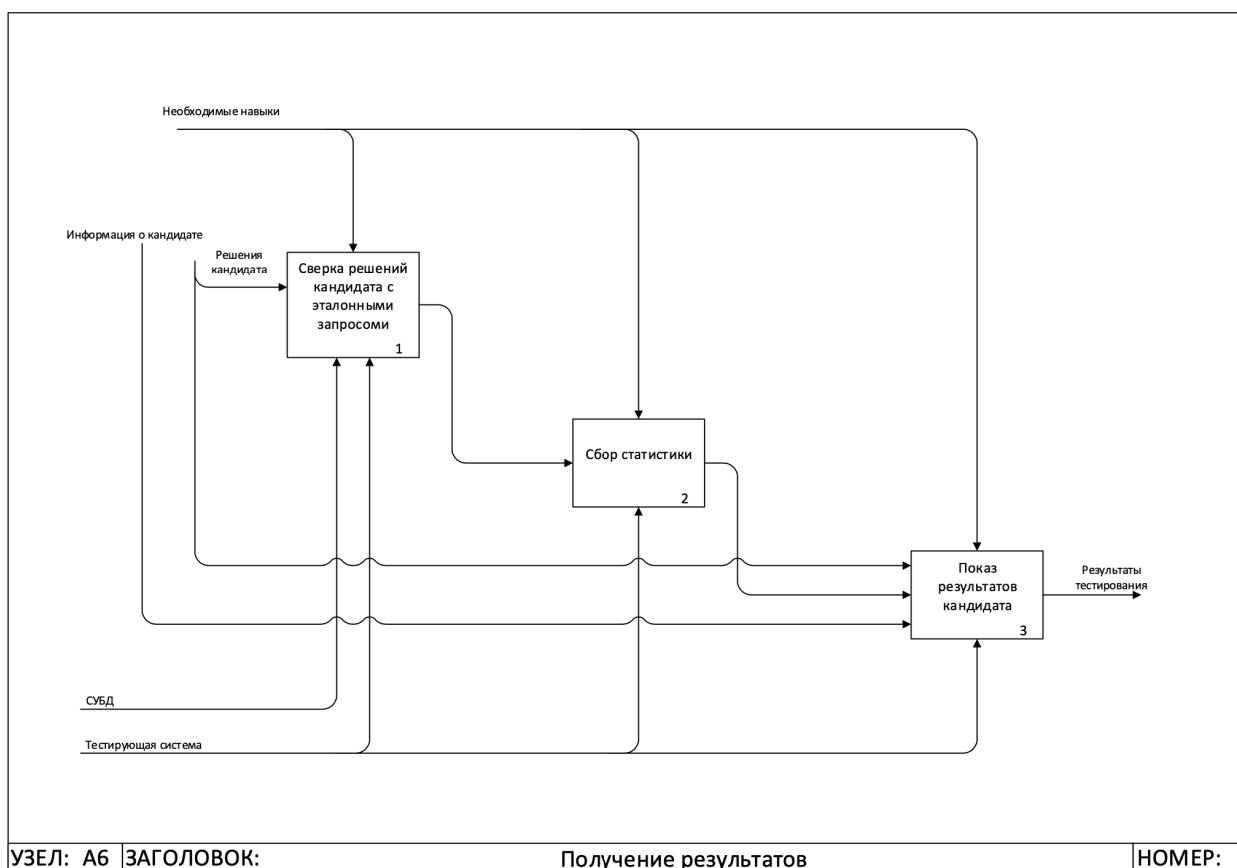


Рисунок 7 – Диаграмма А6. Получение результатов

1.2 Функциональная модель с точки зрения кандидата

С точки зрения кандидата разрабатываемая система намного проще, чем с точки зрения ресурс-менеджера, так как соискателю необходимо лишь пройти уже составленный тест, который был получен по электронной почте.

Кандидат имеет на вход письмо от работодателя с ссылкой на индивидуальный тест в тестирующей системе. Механизмом проведения тестирования является непосредственно разрабатываемая система. В качестве управляющих данных кандидат имеет собственные знания в рассматриваемой предметной области. Выходными данными считаются результаты тестирования, которые будут получены в устной или письменной форме от ресурс-менеджера на собеседовании или по электронной почте соответственно. Полученная начальная контекстная диаграмма представлена на рисунке 8.

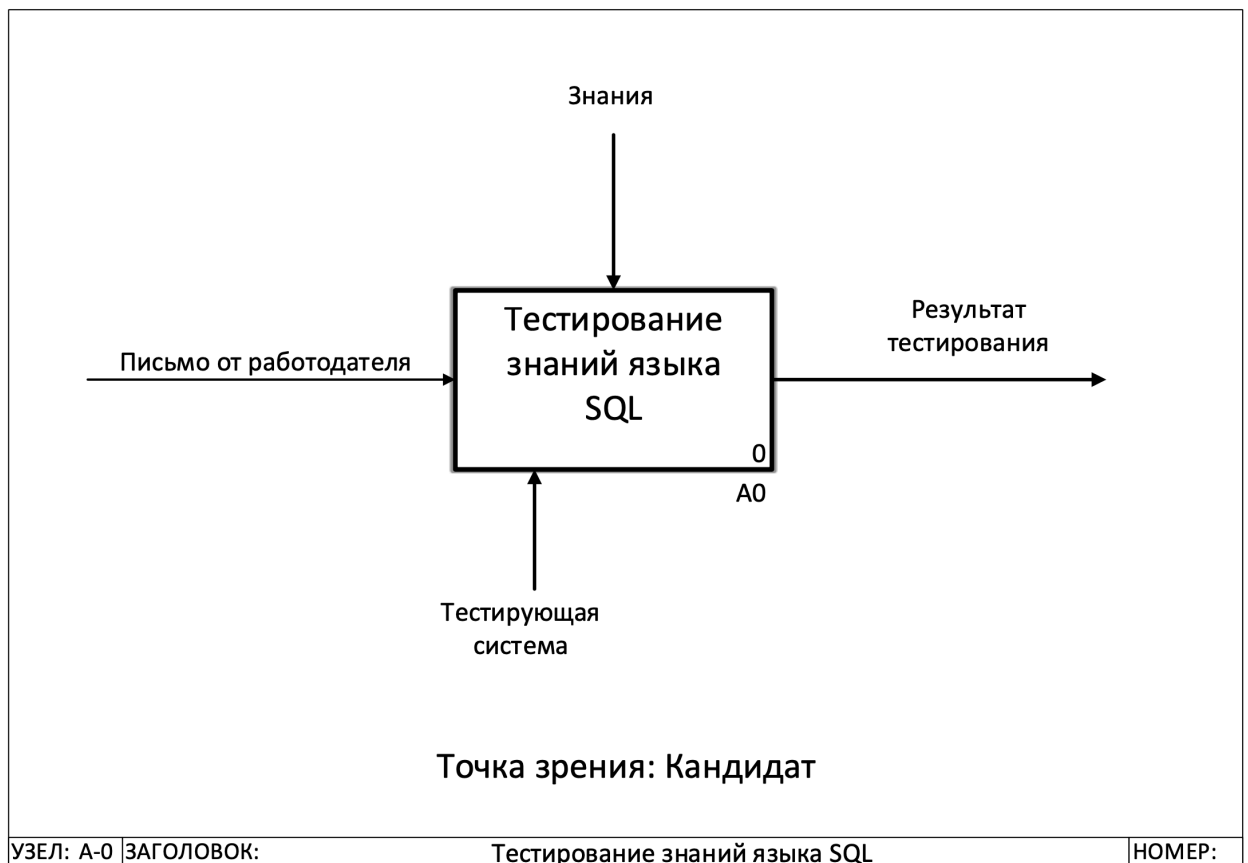


Рисунок 8 – Начальная контекстная диаграмма A-0 с точки зрения кандидата

Представленная начальная контекстная диаграмма может быть детализирована до следующих функциональных блоков:

1. Переход по ссылке из письма.
2. Прочтение инструкций к тестированию.
3. Прохождение тестирования.
4. Сохранение ответов.
5. Получение результатов после рассмотрения ответов менеджером.

Описанная детализация представлена на рисунке 9. Ни один из описанных выше функциональных блоков не может быть более детализирован.

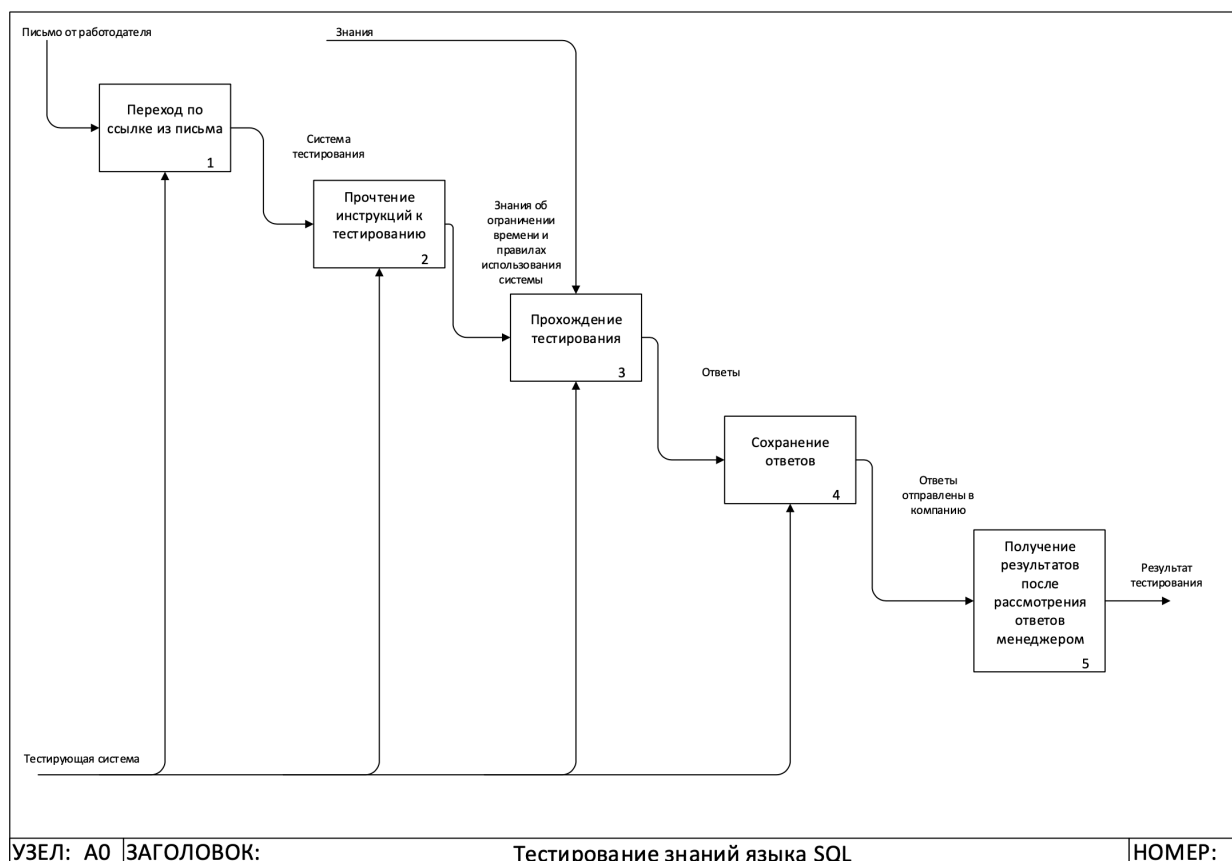


Рисунок 9 – Диаграмма A0. Тестирование знаний языка SQL с точки зрения кандидата

2 Проектирование структуры базы данных

СКЗБД должна содержать объекты, отвечающие различным бизнес-потребностям и использующиеся для различных целей: часть объектов необходимы системе для ее функционирования (системные объекты) и объекты для тестирования (тестовые объекты). Из соображений безопасности имеет смысл разместить различные типы объектов в различных базах данных, в ином случае существует вероятность использования sql инъекций, которые могут принести непоправимый вред системе. Кроме того, тестовые объекты могут находиться в различных СУБД, однако системные объекты должны быть доступны разрабатываемой системе постоянно и находиться в одном месте. Размещение тестовых и системных объектов в различных базах способно решить и эту проблему.

Алгоритм работы с тестовыми объектами будет описан в последующих главах, в данной главе рассмотрим системную базу данных.

2.1 Концептуальное проектирование

Концептуальное проектирование состоит в построении информационной модели самого высокого уровня. Так как бизнес-процессы разрабатываемой системы были описаны с помощью нотации IDEF0, то могут быть получены основные сущности БД:

- Пользователи (Users) – в таблице будут храниться пользователи (ресурс-менеджеры) с логинами и паролями, информация о времени последней авторизации.
- Департамент (OrgUnit) – в таблице будут храниться департаменты компании, в которых может находиться пользователь.
- Схемы (Schemes) – названия схем тестовых объектов с указанием СУБД, создателя схемы и описанием, заполненным создателем схемы.
- Задания (Tasks) – текст задания, дополнительная информация о нем, эталонное решение, информация об используемой схеме.
- Метки (Tags) – таблица с метками которые могут быть использованы в заданиях.
- Метки задания (TaskTags) – связь заданий с метками.
- Комплекты заданий (Packs) – описание комплекта заданий, информация о времени создания комплекта и его создателе, количестве заданий.
- Таблица для связи комплектов и заданий (TasksOfPacks) – таблица будет использоваться для хранения информации о заданиях, которые находятся в комплекте заданий, их порядковом номере в комплекте.
- Отправки комплектов заданий (PacksSendings) – информация об отправках комплектов заданий на электронные почты кандидатов
- История изменений (ChangeHistory) – таблица для логирования любой информации об изменениях.

- Результаты решений (TestResults) – результаты тестирования кандидатов: запрос кандидата и результаты его проверки.

Все описанные сущности вместе со связями могут быть представлены в виде инфологической схемы, представленной на рисунке 10.

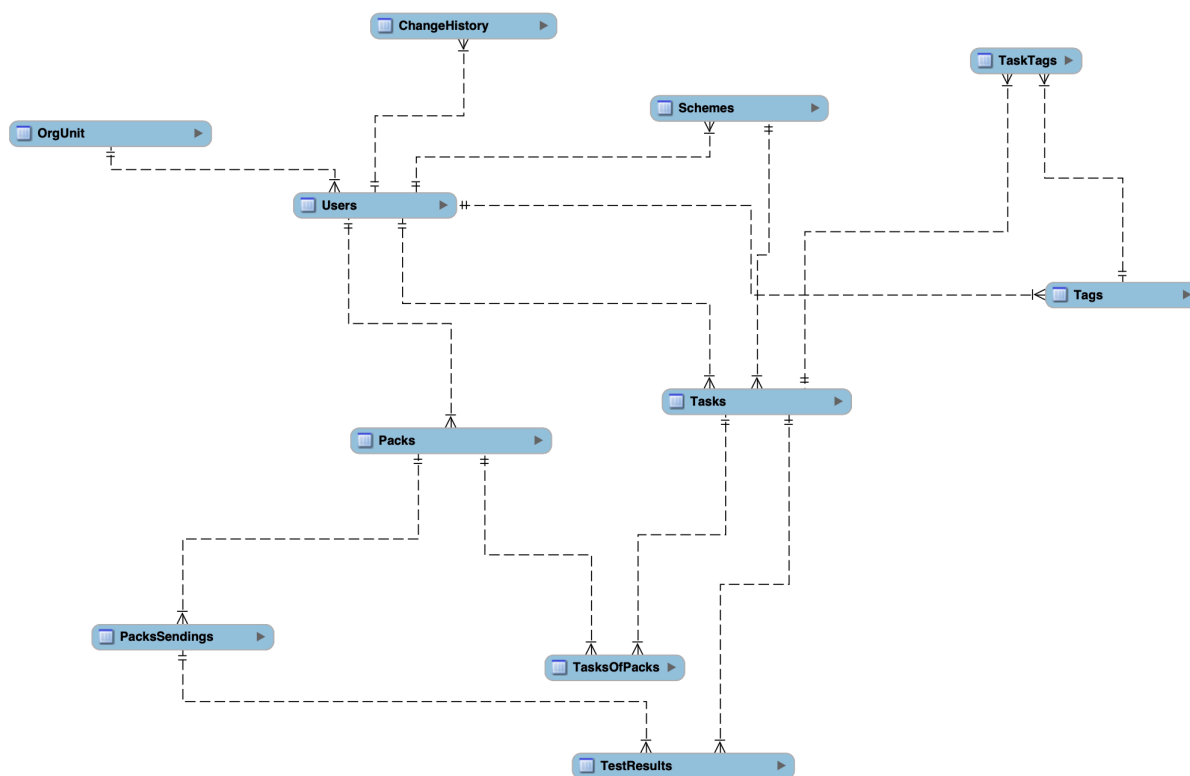


Рисунок 10 – Инфологическая модель системной базы данных

2.2 Логическое проектирование

Этап логического проектирования для данной базы данных состоит в подробном описании полей таблиц, ограничителей целостности. Дatalogическая модель базы данных представлена на рисунке 11.

Большая часть полей соответствует техническому заданию и не нуждается в дополнительных комментариях. Далее будут описаны лишь поля, цель использования которых невозможно понять из ТЗ и бизнес-процессов системы:

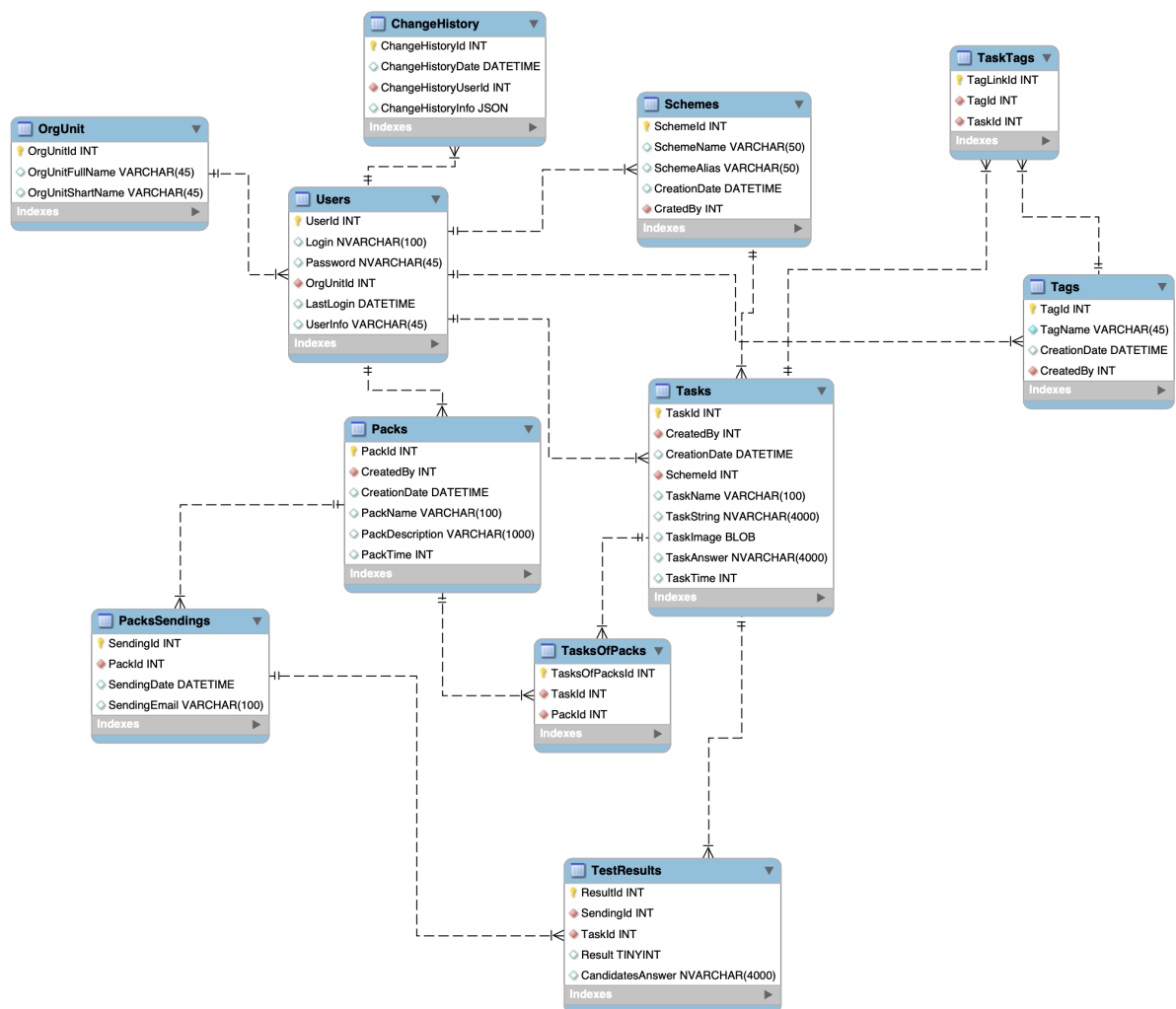


Рисунок 11 – Инфологическая схема системной базы данных

- **CreationDate (DATETIME)** – дата создания задания. Будет использоваться для указания даты и времени создания задания в системе, для понимания актуальности задания.
- **CreatedBy (User)** – пользователь, создавший задание. Может быть использовано для личного контакта при необходимости.
- **ChangeHistoryInfo (JSON)** – любые изменения в системе, которые должны логироваться. Хранится в формате JSON в связи с тем, что изменения могут быть какие угодно и их структура может быть независимой друг от друга.
- **SchemeName (NVARCHAR(50))** – имя схемы базы данных для тестовых объектов для прямого обращения к данной схеме.

- SchemeAlias (NVARCHAR(50)) – пользовательское имя схемы базы данных, указывается при создании.
- RackTime (INT) – общее время на выполнение комплекта заданий. Определяется как сумма времени на выполнение всех заданий.

Основными ограничителями целостности являются первичные и вторичные ключи, которые представлены на даталогической схеме (рисунок 11). В качестве первичных ключей в системе используются автоинкрементируемые суррогатные ключи, так как они позволяют со 100% вероятностью обеспечить целостность данных. Поля, которые являются бизнес ключами будут иметь ограничитель целостности UNIQUE, что позволит контролировать их уникальность без риска нарушения целостности данных.

Работа с базой данных из разрабатываемой системы будет осуществляться в пределах транзакций, что позволит исключить вероятность лишних блокировок и позволит поддерживать данные всегда в актуальном состоянии.

2.3 Физическое проектирование

Основная задача физического этапа проектирования в разрабатываемой базе данных – выбор СУБД.

На основе технического задания система должна работать с системами Oracle 11g или выше и MS SQL Server 2012 и выше. Сравним основные преимущества обеих СУБД.

Преимущества MS SQL Server:

- Простота использования
- Утилиты (SQL Server Profiler, SQL Server Management Studio, BI tools и Database Tuning Advisor)
- Онлайн поддержка

- Простота и скорость восстановления (в сравнении с другими СУБД)
- Цена

Преимущества Oracle:

- Возможность обновлений без перезагрузок
- Кроссплатформенность
- Скорость работы с большими объемами данных (в сравнении с другими СУБД)
- Средства виртуализации

В целом, данные СУБД похожи друг на друга и имеют лишь небольшие различия, не имеющие значения для не высоконагруженных систем. В связи с этим выбор следует делать из соображений квалификации проектной команды. По данному критерию выбор будет сделан в пользу MS SQL Server.

Так как система не является высоконагруженной, то использование дополнительных индексов и каких-либо группировок данных не требуется. Снятие резервных будет производиться средствами системы управления базами данных и операционной системы в автоматическом режиме.

ЗАКЛЮЧЕНИЕ

В ходе преддипломной практики была спроектирована часть системы контроля знаний языка запросов к базам, а именно база данных. Для данной цели были проанализированы бизнес-процессы разрабатываемой системы с помощью нотации IDEF0. На основе этих данных были сделаны выводы о необходимых объектах базы данных для корректной работы разрабатываемой системы.

Проектирование базы данных происходило в 3 основных этапа:

- Концептуальный, который заключался в выделении основных сущностей азы данных и их связей между собой
- Логический – состоял в определении типов данных и колонок таблиц. Кроме того, в данном разделе была затронута тема ограничителей целостности.
- Физический – выбор СУБД и дополнительных технологий для оптимизации работы базы данных

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Основы SQL [Электронный ресурс] <https://proghub.ru/t/sql-basic> (дата обращения 20.02.2020).
2. GeekBrains – образовательный портал [Электронный ресурс] <https://geekbrains.ru/> (дата обращения 20.02.2020).
3. Проектирование баз данных [Электронный ресурс] https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B5%D0%BA%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_%D0%B1%D0%B0%D0%B7_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85 (дата обращения 20.02.2020).
4. Oracle Vs. SQL Server: Key Differences [Электронный ресурс] <https://www.guru99.com/oracle-vs-sql-server.html> (дата обращения 20.02.2020).
5. Oracle и Microsoft SQL Server: прошлое, настоящее и будущее [Электронный ресурс] <https://compress.ru/article.aspx?id=11256> (дата обращения 20.02.2020).
6. СУБД (мировой рынок) [Электронный ресурс] [http://www.tadviser.ru/index.php/%D0%A1%D1%82%D0%B0%D1%82%D1%8C%D1%8F:%D0%A1%D0%A3%D0%91%D0%94_\(%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%BE%D0%B9_%D1%80%D1%8B%D0%BD%D0%BE%D0%BA\)](http://www.tadviser.ru/index.php/%D0%A1%D1%82%D0%B0%D1%82%D1%8C%D1%8F:%D0%A1%D0%A3%D0%91%D0%94_(%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%BE%D0%B9_%D1%80%D1%8B%D0%BD%D0%BE%D0%BA)) (дата обращения 20.02.2020).
7. IDEF0 [Электронный ресурс] <https://ru.wikipedia.org/wiki/IDEF0> (дата обращения 20.02.2020).
8. Знакомство с нотацией IDEF0 и пример использования [Электронный ресурс] <https://habr.com/ru/company/trinion/blog/322832/> (дата обращения 20.02.2020).