Добрый день уважаемая комиссия, представляю вам проект, который я разработал – «Графовая базы данных для системы управления проектами».

Поскольку каждая из существующих систем управления проектами предлагает свой набор инструментов и сервисов для удобной и эффективной работы над проектами и зачастую в рамках одного проекта используется несколько таких систем, становится актуальной задача их интеграции.

Выбор пал именно на граф ориентированную модель данных, поскольку данные проекта тесно связаны между собой в отношениях и могут углубляться в несколько уровней. Также хотелось отойти от обыденного реляционного подхода где таблицы имеют строго заданные поля и строгую схему. Было необходимо разработать гибкую структуру данных, которая давала бы возможность изменять себя по мере появления требований к интегрированной системе и что немало важно, разработанная база данных должна предоставлять высокую производительность при работе со связанными данными.

**Слайд 2. Цель и задачи ВКР**

Целью работы является разработка графовой базы данных для системы управления проектами. Для достижения этой цели были поставлены следующие задачи:

1. Разработка графовой модели данных для системы управления проектами с целью интеграции данных о проектах из существующих систем;
2. Создание тестовых скриптов для заполнения графовой базы данных;
3. Создание запросов для извлечения информации о проектах с целью проверки работоспособности разработанной модели;
4. Уменьшение времени отклика на запросы пользователей интегрированной системы за счет использования графовой модели данных и СУБД Neo4j.

**Слайд 3. Интегрируемые системы управления проектами**

Исходя из предложенного перечня систем управления проектами, в графовую базу данных интегрируются данные о проектах таких систем как:

1. GitHub – крупнейший веб сервис для совместной разработки проектов основанный на системе контроля версий Git.
2. Bitbucket – распределенная система контроля версий, созданная для профессиональных команд. Единственное Git-решение, которое хорошо масштабируется и подходит для команд любых размеров;
3. Jira – это коммерческая, настраиваемая система отслеживания ошибок для управления проектами. Поддерживает командную работу в режиме реального времени, предоставляет высокий уровень безопасности, поддерживает email-интеграцию и русский интерфейс;
4. GitLab – является веб-менеджером репозитория Git, предназначенным для написания кода, его развертывания и тестирования.

Перечисленные системы управления проектами базируются на использовании репозитория файлов Git.

Git – это система контроля версий (VCS) для отслеживания изменений в компьютерных файлах и координации работы с этими файлами среди множества людей.

**Слайд 4. Преимущества графовой СУБД Neo4j**

Поскольку разработка графовой базы данных осуществлялась на СУБД Neo4j, хочу отметить некоторые ее преимущества, а именно:

1. Гибкость, которая дает возможность хранить одни и те же объекты с различными свойствами, что является удобным при интеграции систем;
2. Ощутимый прирост производительности при работе со связанными данными;
3. Возможность расширения без нарушения существующих запросов и функционала приложения или системы;
4. Графический интерфейс визуализации данных;
5. Собственный декларативный графовый язык запросов Cypher, который позволяет выразительно и эффективно строить запросы и обновлять графовые хранилища;

**Слайд 5. Разработка модели графовой базы данных**

Разработанная графовая база данных для системы управления проектами хранит такие данные как:

1. наименования и описания проектов, их заказчиков и руководителей;
2. наименования и описания задач, подзадач проекта;
3. исполнителей (разработчиков) назначенных на задачи;
4. совершённые изменения и их описание на этапе выполнения задач;
5. URL адреса репозиториев интегрируемых систем в которых были созданы задачи или совершены их изменения на этапе выполнения.

**Слайд 6. Схема разработанной графовой БД**

Схема разработанной графовой базы данных выглядит следующим образом (рисунок схемы).

Вершины графовой базы данных хранят такие данные как:

* ФИО заказчиков (Customer);
* наименования проектов (Project);
* статусы проектов (Project\_status);
* типы проектов (Project\_type);
* коды проектов (Project\_code);
* ФИО руководителей проектов (Master);
* наименования задач (Issue);
* типы задач (Issue\_type);
* приоритеты задач (Issue\_priority);
* статусы задач (Issue\_status);
* ФИО разработчиков (Developer);
* совершённые изменения и их описание (Commit);
* наименования и URL адреса репозиториев интегрируемых систем (Repository).

Отношения между вершинами, позволяют понять, как связаны между собой данные в разработанной модели.

Далее, я привел несколько запросов и их результатов выполнения что бы продемонстрировать работу базы данных.

**Слайд 7. Результаты запросов**

Исходя из разработанной мною схемы графовой БД, у проектов есть свои заказчики и руководители, они разделены на задачи, а также в БД хранятся сведения о кодах, статусах и типах проектов. Для получения такой информации, строится запрос, представленный на слайде.

**Слайд 8. Результаты запросов**

В моей графовой структуре, задача характеризуются своим статусом, приоритетом, типом, разработчиком, выполняющим задачу и репозиторием где задача была создана. Для получения такой информации был создан следующий запрос.

**Слайд 9. Результаты запросов**

При управлении проектами, важной информацией является история изменений задач, то как проходит их выполнение и на какой стадии находятся назначенные разработчики. Для получения истории изменений конкретной задачи, имени разработчика, а также репозитория где совершались изменения, строится следующий запрос.

**Слайд 10. Охрана труда**

В виду того, что взаимодействие с разработанной графовой базой данных будут осуществлять системные администраторы, была произведена оценка условий труда системного администратора по факторам тяжести и напряженности трудового процесса до и после разработки. Некоторые показатели тяжести труда улучшились, но класс условий труда не изменился и остался в пределах «Допустимого». Разработанная графовая база данных для системы управления проектами не повлияла на показатели напряженности трудового процесса.

**Слайд 11. Обоснование экономической эффективности ВКР**

Для обоснования экономической эффективности было учтено время, затраченное на выполнение выпускной квалификационной работы, которое составило 40 дней.

При расчете стоимости работы, учитывались расходы на материалы, связь, источники информации и расходы на электроэнергию. Определены расходы на оплату труда дипломника, дипломного руководителя и консультантов. Также были рассчитаны затраты на амортизацию оборудования. Общая стоимость ВКР составила 13 688,91 руб.

Экономия средств при использовании данной разработки составляет 15 000 руб./год.

Период окупаемости разработки составил приблизительно 11 месяцев.

**Выводы**