

Effortly-Time-Tracker

Android application for conveniently measuring time for everyday or work tasks

useful link

[my er](#)

[ER habr](#)

[ER DIAGRAMS](#)

[use case DIAGRAMS](#)

tz

Разработка базы данных для хранения и обработки данных для приложения учета и аудита времени, потраченного на рабочие и личные задачи.

Сформулировать описание пользователей проектируемого приложения по учету и аудиту времени, потраченного на рабочие и личные задачи для доступа к базе данных

Спроектировать сущности базы данных и ограничения целостности учета и аудита времени, потраченного на рабочие и личные задачи.

Выбрать средства реализации базы данных и приложения (в том числе выбор СУБД).

Провести исследование зависимости времени выполнения запросов от количества записей в базе данных.

todo

- актуальность
- выписывать интерфейс

main

В рамках курсовой работы должны быть решены две основные задачи:

- проектирование и разработка базы данных
- разработка приложения (или приложений) доступа к базе данных

Работа над курсовым проектом состоит из четырех частей, соответствующих четырем основным разделам РПЗ курсовой работы. Основные этапы работы над курсовой работой:

- аналитический раздел:

- уточнение требований и формулировка ограничений предметной области;
- анализ существующих решений;

- конструкторский раздел:

- проектирование базы данных;
- проектирование способов взаимодействия с базой данных;

- технологический раздел:

- разработка базы данных и приложения доступа к базе данных;
- тестирование разработанного приложения;

- исследовательский раздел:

- проведение исследования разработанной базы данных.

Требования к проектируемой базе данных

Обязательными требованиями к проектируемой базе данных являются:

- Наличие не менее 7 сущностей в базе данных;
- Наличие связей между сущностями базы данных (не менее одной связи многие-ко-многим);
- Наличие ролевой модели на уровне базы данных (не менее трех ролей);
- Наличие протестированной хранимой процедуры или функции на стороне базы данных

Общие требования

Сроки

Основные этапы выполнения курсовой работы:

- 2 неделя - прием пожеланий по научным руководителям. Прием пожеланий организован с использованием Яндекс Формы. По окончании второй недели прием пожеланий завершается, формируется распределение студент-руководитель Ссылка на форму - <https://forms.yandex.ru/cloud/63e7593ceb61465b0e8159a1>
- 3 неделя - прием тем курсового проекта/ прием ТЗ. Прием ТЗ организован с использованием Яндекс Формы. По окончании третьей недели прием пожеланий завершается, формируются списки студент-руководитель-тема. Если обратная связь от студента не получена, выдача руководителя и темы производится на усмотрение кафедры. Ссылка на форму - <https://forms.yandex.ru/cloud/63e7e62543f74f6e05ec29c9>
- 8 неделя - смотр (заочный) курсовых работ, включающий в себя проверку аналитического и конструкторского разделов. По результатам смотра студенты получают общий список замечаний к курсовым работам
- 13 неделя - смотр (заочный) курсовых работ, включающий в себя проверку технологического и исследовательского разделов. По результатам смотра студенты получают общий список замечаний к курсовым работам Прием курсовых работ проводится:
- На 15 и 16 неделях семестра;
- В случае уважительных причин возможна организация дополнительных дней сдачи с предварительным уведомлением студентов на 17 неделе семестра.

Оценка курсовой работы

Для сдачи курсовой работы необходимо иметь следующий набор документов:

- Второй экземпляр ТЗ, подписанный научным руководителем и заведующим кафедрой.

о В конце 4-ой недели студенты получают ТЗ в электронном виде. ТЗ формируется по данным, отправленным в форму для регистрации или выдается кафедрой в случае их отсутствия (Подача пожеланий по теме в форме “Курсовой проект по дисциплине Базы Данных” сданным ТЗ не считается);

о Студент печатает ТЗ и подписывает его сам и у научного руководителя;

о В конце 14 недели студент лично получает у ответственного за курсовую работу преподавателя экземпляр ТЗ, подписанный заведующим кафедрой, в обмен на

ТЗ, подписанное самим студентом и научным руководителем.
После чего
подписывает ТЗ у научного руководителя (можно совместить с
подписью РПЗ) и
вшивает в РПЗ.

- Расчетно-пояснительная записка не менее 20 листов, подписанная научным руководителем. На титульном листе также должна присутствовать рекомендуемая научным руководителем оценка курсовой работы.
- Презентация к курсовой работе в двух экземплярах. Один экземпляр подшивается в РПЗ, второй сдается приемной комиссии. Для защиты курсовой работы также допустим электронный формат презентации. 5
- Разработанный программное обеспечение.
- Заключение о пройденном нормоконтроле.

Следующие пункты описывают возможные ситуации по снижению оценки:

- Несвоевременная сдача курсовой работы;
- Отсутствие или ненадлежащая реализация одной или нескольких частей курсового проекта

Аналит

Объектно-ориентированный анализ и проектирование

Основная идея объектно-ориентированного подхода состоит в рассмотрении предметной области и логического решения задачи с точки зрения объектов (понятий и сущностей). В процессе объектно-ориентированного анализа основное внимание уделяется определению и описанию объектов (или понятий) в терминах предметной области. В процессе объектно-ориентированного проектирования определяются логические программные объекты, которые будут реализованы средствами объектно-ориентированного языка программирования. Эти программные объекты включают в себя атрибуты и методы. И, наконец, в процессе конструирования или объектно-ориентированного программирования обеспечивается реализация разработанных компонентов и классов. Основные принципы объектно-ориентированного анализа и проектирования сводятся к следующему:

1. Анализ требований, во время которого выделяются основные процессы, происходящие в моделируемой системе и их формулировка в виде прецедентов. Прецедент – это текстовое описание процессов, происходящих в предметной области.

2. Объектно-ориентированный анализ предметной области. Задача этого шага в определении видов деятельности участников процесса и составлении концептуальной модели, которая отражает различные категории элементов предметной области. Причем не только виды деятельности участников, но и все относящиеся к делу понятия.
3. Объектно-ориентированное проектирование, при котором основное внимание сосредоточено на распределении обязанностей. Распределение обязанностей означает выделение задач и обязанностей различных программных объектов в приложении. Наиболее важным моментом объектно-ориентированного анализа и проектирования является квалифицированное распределение обязанностей между компонентами программной системы. Обязанности объектов и их взаимодействия изображаются с использованием диаграмм классов и диаграмм взаимодействий. Основным инструментальным средством объектно-ориентированного анализа и проектирования является унифицированный язык моделирования UML (Unified Modeling Language). Технология UML, одобренная консорциумом OMG (Object Management Group), является мощным средством описания бизнес-процессов и представления их в той форме, которая устраивает как разработчиков, так и пользователей. В распоряжении проектировщика баз данных имеется множество UML-диаграмм (<http://www.uml.org>), которые часто используются на этапе определения требований, развертывания и на всех промежуточных этапах проектирования.

Приступая к работе, нужно добиться простого и ясного видения проблемы, при котором задачи и приоритеты проекта стали бы очевидными. Атрибутом хорошего видения проблемы является центральная идея (лейтмотив проекта).

мое решение видения проблемы

Шаг 1: Определение проблемы

Проблема должна быть актуальной и понятной для целевой аудитории.

"Многие люди и организации сталкиваются с трудностями в эффективном управлении временем из-за отсутствия инструментов для точного учета и анализа затраченного времени на задачи"

Шаг 2: Центральная идея проекта

Это должно быть краткое утверждение, отражающее основную цель и предлагаемое решение.

"Разработка удобного в использовании приложения, которое позволяет пользователям точно учитывать время, затраченное на задачи, и предоставляет аналитику для оптимизации управления временем".

Когда требования сформулированы, необходимо проанализировать их в целом и каждое по отдельности. Каждому требованию должен быть назначен тот или иной приоритет. Детализация приоритетов может быть любой, в зависимости от потребностей. Уровни приоритета можно определить следующим образом:

1. **Необходимые.** Эти требования обязательно должны быть воплощены в программе, без этого ее нельзя выпускать на защиту. Необходимые функции должны быть реализованы и испытаны как можно раньше.
2. **Желательные.** Присутствие этих требований крайне желательно. При достаточном обосновании от их реализации можно отказаться, но это не уменьшает важности их наличия в программе. Реализация и испытания желательных требований начинается сразу после обязательных.
3. **Возможные.** Эти требования также желательны, но реализуются в последнюю очередь и являются первыми кандидатурами на удаление, если вдруг возникнут проблемы со сроками. Обычно проект начинается с создания документа, содержащего описание его основных целей. В описании формулируются очень абстрактные (и зачастую нетехнические) требования к системе, необходимые для понимания того, что же собой представляет система в целом.

мое решение

1.Необходимые требования

Эти требования являются фундаментальными для функционирования системы и должны быть реализованы в первую очередь. Для проекта по учету времени к таким требованиям могут относиться:

- Возможность регистрации и аутентификации пользователей. (google)
- Функционал создания, редактирования и отслеживания задач и проектов. (статичные карточки)
- Функционал создания и изменения todo list
- Механизм замера времени и привязки его к конкретным задачам.
- Основные отчеты по затраченному времени для индивидуального пользователя.

для админа доп от пользователя :

- создание и редактирование групп пользователей

для аудита доп от пользователя :

- аудит аналитики
- сохранение аналитики

2. Желательные требования

Желательные требования улучшают пользовательский опыт и функциональность системы, но от их реализации можно отказаться без критического ущерба для проекта. Примеры таких требований могут включать:

- для пользователя:
 - Расширенные аналитические отчеты с графиками и диаграммами.
 - добавить интеграцию с гугл календарем
 - Уведомления
 - заметки
 - создать карточки и прикольная расстановка карточек как в канване
- для админа:
 - ставить принадлежность задаче определнного пользователя группы
 - Мониторинг активности и производительности группы:
 - Управление доступом: Функции для детализированного управления доступом к проектам и задачам на основе ролей и прав пользователей улучшат безопасность и эффективность совместной работы.
- для аудита:
 - доп аналитика
 - История изменений

3. Возможные требования

Возможные требования могут быть добавлены в проект, если это позволяют ресурсы и время. Они могут включать функции, которые не являются критически важными, но могут значительно улучшить пользовательский опыт или предоставить дополнительную ценность. Примеры:

- Для пользователя:
 - Возможность экспорта данных в различные форматы (например, CSV, PDF).
 - добавление доп таймеров типа помидора
 - добавить функции выработки привычек
 - мотивационные штучки
 - более гибкое редактирование пространства приложения
 - Многоязычность
 - Резервное копирование и восстановление:
 - Отзывы и обратная связь:
- Для админа:
 - добавление различных ролей у групп
 - добавление у группы групп и назначение мини админов у них

- для аудита:
 - доп аналитика с нейронкой

доп инфа

Обычно проект начинается с создания документа, содержащего описание его основных целей. В описании формулируются очень абстрактные (и зачастую нетехнические) требования к системе, необходимые для понимания того, что же собой представляет система в целом.

В рамках любой предметной области действует широкий набор политик, законов и промышленных стандартов. Все эти контролирующие принципы в целом называются правилами делового регламентами или деловыми правилами. **Правила делового регламента** - один из основных источников функциональных требований, поскольку они определяют возможности, которыми должна обладать система для выполнения правил. Для создания и документирования правил делового регламента и их применения разработаны различные методики, а для организации правила делового регламента предложено множество разных таксономий (схем классификаций). Простейшая из них включает пять типов правил.

1. Факты (или инварианты) – это неизменные истины о сущности данных и их атрибутах. Зачастую они описывают связи и отношения между важными бизнес-терминами. Примеры фактов: стоимость билетов не возвращается, если покупатель изменяет маршрут; со стоимости доставки налог с продаж не берется.
2. Ограничения определяют, какие операции может выполнять система и ее пользователи. Вот некоторые слова и фразы, которые часто применяются при описании ограничивающего правила: должен, не должен, не может и только. Примеры ограничений: а. постоянный посетитель библиотеки может отложить для себя до 10 книг; б. экипажи коммерческих авиарейсов должны каждые 24 часа отдыхать не менее 8 часов;
3. Активаторы операций – это правила, при определенных условиях приводящие к выполнению каких-либо действий. Выразительным способом документирования активирующих операции являются таблицы решений. Примеры активаторов операций: а. если срок хранения контейнера с химикатом истек, об этом необходимо уведомить лицо, у которого в данный момент находится контейнер; б. если клиент заказал книгу автора, написавшего несколько книг, клиенту следует предложить другие книги этого автора, прежде чем принять заказ,
4. Вывод - это правило, устанавливающее новые реалии на основе достоверности определенных условий. Вывод создает новый факт на основе других фактов или вычислений. Выводы зачастую записывают в формате «если — то», применяемом также при записи правил, активирующих операции; тем не менее, раздел «то» вывода включает в себе факт или предположение, а не

действие. Примеры выводов: а. если платеж не поступил в течение 30 календарных дней с момента отправки счета, счет считается просроченным; b. если поставщик не может поставить заказанный товар в течение пяти дней с момента получения заказа, заказ считается невыполненным;

5. Вычисления – это один из классов правил, которые определяют вычисления, выполняемые с использованием математических формул и алгоритмов. В отличие от активирующих операций правила вычислений можно рассматривать в качестве требований к программному обеспечению: а. цена единицы товара снижается на 10% при заказе от 6 до 10 единиц, на 20% - при заказе от 11 до 20 единиц и на 30% - при заказе свыше 20 единиц; b. общая стоимость заказа вычисляется как сумма стоимостей всех заказанных товаров, за вычетом скидок на количество, плюс государственные и местные налоги, действующие в округе, куда будет доставлен товар, плюс стоимость доставки и плюс необязательный страховой сбор. Подробные функциональные и нефункциональные требования к разрабатываемой системе должны быть записаны в спецификации к требованиям. Существует несколько способов представления требований:
6. документация, в которой используется четко структурированный и аккуратно используемый естественный язык;
7. графические модели, иллюстрирующие процессы трансформации, состояния системы и их изменения, взаимодействия данных, а также логические потоки, классы объектов и отношения между ними;
8. формальные спецификации, где требования определены с помощью математически точных, формальных логических языков. После формулирования требований проводят их анализ и осуществляют построение концептуальной и логической модели информационной системы, в соответствии с выбранной методологией. Подготавливается набор диаграмм, одной из которых, вероятно, будет ER-диаграмма (в случае структурного подхода)

Требования к аналитической части курсовой работы

- Анализ предметной области. Сравнительный анализ существующих "конкурентов";
- Формулировка требований к разрабатываемой базе данных и проложению;
- Формализация и описание информации, подлежащей хранению в проектируемой базе данных;
- Проведенный анализ существующих баз данных на основе формализации данных;
- ER-диаграмма сущностей проектируемой базы данных в нотации Чена;
- Формализация и описание пользователей проектируемого приложения к базе данных;
- Диаграмма вариантов использования.

Роли

Роль пользователя

0. регистрация и аутентификация;

1. Добавление :

- личных проектов;
- личных задач;
- личных тегов;
- записи в todo-list;

2. Изменение или удаление:

- личных проектов;
- личных задач;
- личных тегов;
- записи в todo-list;

3. Просмотр:

- личных проектов;
- личных задач;
- личных тегов;
- todo-list;

4. Поиск:

- личных проектов;
- личных задач;
- личных тегов;

5. Замер времени выполнение задач

Роль админ (в группе)

0. регистрация и аутентификация;

1. Добавление :

- личных проектов;
- личных задач;
- личных тегов;
- записи в todo-list;

2. Изменение или удаление:

- личных проектов;

- личных задач;
- личных тегов;
- записи в todo-list;

3. Просмотр:

- личных проектов;
- личных задач;
- личных тегов;
- todo-list;

4. Поиск:

- личных проектов;
- личных задач;
- личных тегов;

5. Замер времени выполнения задач

6. Создание группы:

- добавление имени и изменение;

7. редактирование группы:

- добавление пользователей;
- удаление пользователей;

Роль аудит

0. регистрация и аутентификация;

1. Добавление :

- личных проектов;
- личных задач;
- личных тегов;
- записи в todo-list;

2. Изменение или удаление:

- личных проектов;
- личных задач;
- личных тегов;
- записи в todo-list;

3. Просмотр:

- личных проектов;

- личных задач;
- личных тегов;
- todo-list;

4. Поиск:

- личных проектов;
- личных задач;
- личных тегов;

5. Замер времени выполнения задач

6. просмотр аналитики группы

7. сохранение аналитики группы

Что может это приложение

Приложение по учету и аудиту времени представляет собой комплексный инструмент, разработанный для помощи пользователям в эффективном управлении временем, используемым на повседневные и рабочие задачи. Это решение предназначено как для индивидуальных пользователей, так и для организаций, стремящихся оптимизировать продуктивность и эффективность работы.

Основные возможности приложения включают:

Регистрация и аутентификация пользователей через Google, обеспечивая безопасный доступ к личным и рабочим данным. Управление задачами и проектами с функционалом создания, редактирования и отслеживания прогресса выполнения в удобном статичном карточном интерфейсе. Создание и редактирование todo-list'ов, позволяющее пользователям организовывать ежедневные задачи и дела. Замер времени выполнения задач, предоставляющий точные данные о времени, затраченном на конкретные действия или проекты. Генерация отчетов о времени, предоставляющая индивидуальным пользователям подробный анализ использования времени.

Дополнительные функции для администраторов групп включают:

Создание и редактирование групп пользователей, упрощая управление командами и проектами внутри организации.

Для роли аудита доступны следующие возможности:

Аудит и анализ аналитики по группам, что позволяет глубоко понимать распределение времени среди членов команды.

Сохранение аналитических данных, обеспечивающее возможность отслеживать изменения в эффективности использования времени и корректировать процессы управления.

Это приложение предназначено для тех, кто ищет способы улучшить управление временем, будь то индивидуальные пользователи, желающие повысить свою личную продуктивность, команды, стремящиеся оптимизировать рабочие процессы, или аудиторы, задача которых — анализировать и предоставлять рекомендации по улучшению эффективности использования времени внутри организаций.

old
