



Федеральное государственное автономное образовательное
учреждение высшего образования «Московский физико-
технический институт (национальный исследовательский
университет)»

Практическая работа на тему:
**«Автоматизация учета рабочего
времени сотрудников компании»**

Выполнил студент Б03-113И группы МФТИ
Дойников Андрей Владимирович

Содержание

1. Введение	3
2. Постановка задачи	3
3. Проектирование системы	3
3.1 Выбор технологий и библиотек	3
3.2 Архитектура приложения	4
3.3 Взаимодействие между модулями	4
3.4 Структура базы данных	5
3.5 Пользовательский интерфейс	6
4. Пример работы программы	6
4.1 Управление сотрудниками	6
4.2 Настройка рабочего дня	7
4.3 Учет рабочего времени	8
4.4 Просмотр статистики	10
5. Заключение	10

1. Введение

Разработанная система учета рабочего времени призвана решить проблемы отслеживания времен прихода и ухода сотрудников, предоставляя удобный интерфейс, а также обеспечивая автоматический расчет ключевых показателей, таких как среднее опоздание, средняя переработка и средняя продолжительность рабочего дня.

2. Постановка задачи

Целью разработки является создание программного обеспечения для автоматизации процесса учета рабочего времени сотрудников компании с возможностью ведения статистики.

Основные функциональные требования к системе:

- Добавление новых сотрудников с указанием должности и даты найма
- Удаление сотрудников из системы при необходимости
- Внесение информации о времени прихода и ухода каждого сотрудника
- Установка начала и окончания стандартного рабочего дня
- Расчет и отображение статистики по каждому сотруднику
 - Среднее опоздание
 - Средняя переработка
 - Средняя продолжительность рабочего дня
- Расчет и отображение общей статистики по всем сотрудникам компании

Технические требования:

- Данные должны храниться в базе данных
- Интерфейс должен быть интуитивно понятным и не перегруженным

3. Проектирование системы

3.1 Выбор технологий и библиотек

Было принято решение разрабатывать приложение на языке Python, так как он простой, отлично подходит для быстрой разработки приложения, а также на нем удобно работать с графическим интерфейсом и СУБД.

Для создания графического интерфейса была выбрана библиотека PySide6 благодаря ее кроссплатформенности. Есть более популярные аналоги, например Tkinter и PyQt, но они имеют проблемы при установке на macOS.

Для управления базами данных был выбран SQLite, так как он встроен в Python и не требует отдельной установки.

3.2 Архитектура приложения

Система разработана с использованием модульного подхода и разделена на следующие компоненты:

1. **main.py** - основной файл запуска программы, инициализирующий базу данных и пользовательский интерфейс.
2. **database.py** - модуль для работы с базой данных SQLite, содержащий класс Database с методами для выполнения операций с данными.
3. **models.py** - модуль, определяющий классы моделей данных (Employee, TimeRecord).
4. **ui.py** - модуль, отвечающий за пользовательский интерфейс, содержащий класс MainWindow.

Такая структура обеспечивает четкое разделение ответственности между компонентами системы и облегчает дальнейшую поддержку и расширение функциональности.

3.3 Взаимодействие между модулями

Разработанное приложение следует принципам объектно-ориентированного программирования и модульности. Взаимодействие между компонентами системы организовано следующим образом:

1. **main.py** и **Database**:
 - В файле **main.py** создается экземпляр класса **Database**, который устанавливает соединение с базой данных SQLite
 - При запуске приложения этот экземпляр инициализирует структуру базы данных (если она не существует)
 - После завершения работы приложения метод **close()** экземпляра **Database** корректно закрывает соединение с базой данных
2. **Database** и модели данных:
 - Класс **Database** реализует методы для работы с данными, такие как **add_employee()**, **remove_employee()**, **add_time_record()** и др.
 - Для структурирования данных, получаемых из базы, используются классы **Employee** и **TimeRecord** из модуля **models.py**
 - Метод **from_db_row()** в классах **Employee** и **TimeRecord** преобразует результаты запросов из базы данных в объекты соответствующих классов
3. **MainWindow** и **Database**:
 - При создании экземпляра класса **MainWindow** ему передается экземпляр класса **Database**

- MainWindow хранит ссылку на Database и использует его методы для получения и сохранения данных
 - Когда пользователь выполняет действия в интерфейсе (добавление сотрудника, запись времени и т.д.), MainWindow вызывает соответствующие методы Database
4. MainWindow и модели данных:
- Получая данные из базы через Database, MainWindow преобразует их в объекты классов Employee и TimeRecord
 - Эти объекты используются для отображения информации в пользовательском интерфейсе
 - При обновлении интерфейса (например, при добавлении нового сотрудника) MainWindow обновляет соответствующие элементы управления с использованием данных из этих объектов

Информационные потоки

Запись данных:

Пользовательский ввод → MainWindow → Database → База данных

Чтение данных:

База данных → Database → Модели данных → MainWindow → Элементы интерфейса

3.4 Структура базы данных

База данных SQLite включает три основные таблицы:

1. **employees** - таблица сотрудников со следующими полями:
 - id (INTEGER PRIMARY KEY) - уникальный идентификатор сотрудника
 - name (TEXT) - ФИО сотрудника
 - position (TEXT) - должность сотрудника
 - hire_date (TEXT) - дата найма в формате YYYY-MM-DD
2. **time_records** - таблица учета рабочего времени:
 - id (INTEGER PRIMARY KEY) - уникальный идентификатор записи
 - employee_id (INTEGER) - внешний ключ, связывающий запись с сотрудником
 - date (TEXT) - дата в формате YYYY-MM-DD
 - arrival_time (TEXT) - время прихода в формате HH:MM
 - departure_time (TEXT) - время ухода в формате HH:MM
3. **work_settings** - таблица с настройками рабочего времени:
 - id (INTEGER PRIMARY KEY) - идентификатор (всегда 1, так как таблица содержит только одну запись)

- start_time (TEXT) - время начала рабочего дня в формате HH:MM
- end_time (TEXT) - время окончания рабочего дня в формате HH:MM

3.5 Пользовательский интерфейс

Интерфейс приложения разделен на три основные вкладки:

1. **Сотрудники** - вкладка для управления списком сотрудников, включает таблицу сотрудников и форму для добавления новых сотрудников.
2. **Учет времени** - вкладка для фиксации времени прихода и ухода сотрудников, а также просмотра истории посещений.
3. **Статистика** - вкладка для отображения статистических данных по каждому сотруднику и по компании в целом.

В верхней части окна расположен блок настроек стандартного рабочего дня.

4. Пример работы программы

Рассмотрим пример работы программы. На рис.1 изображён интерфейс программы при её запуске.

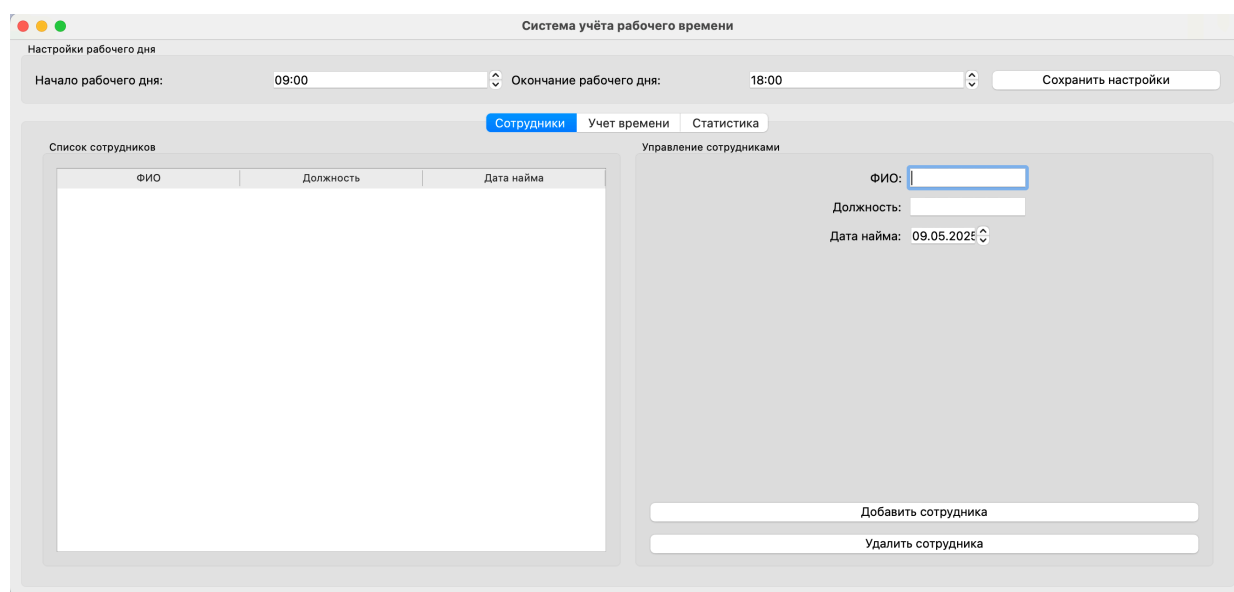


Рис.1 Интерфейс программы при запуске

4.1 Управление сотрудниками

Первый шаг - добавление сотрудников в систему. Для этого на вкладке "Сотрудники" вводим данные в форму: ФИО, должность и дата найма (по умолчанию устанавливается текущая дата). Нажимаем кнопку "Добавить сотрудника".

На рис.2 показан результат добавления нескольких сотрудников. В таблице отображаются ФИО, должность и дата найма каждого сотрудника.

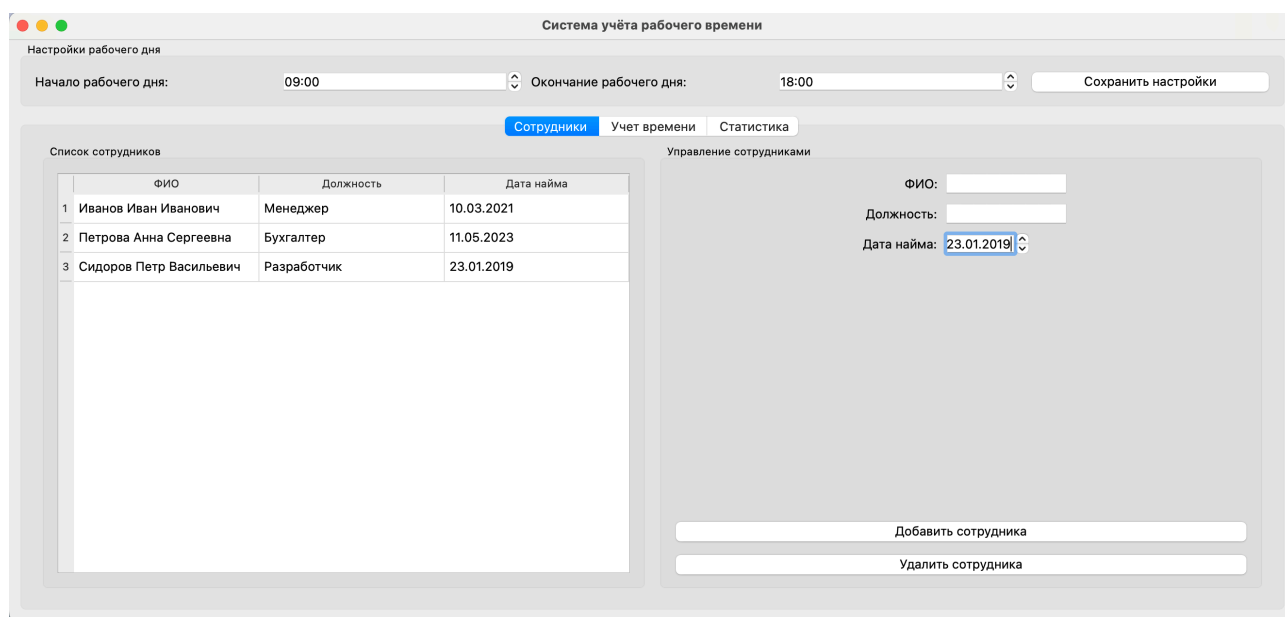


Рис.2: Вкладка "Сотрудники" после добавления нескольких сотрудников

Для удаления сотрудника выбираем его в таблице и нажимаем кнопку "Удалить сотрудника". Система запрашивает подтверждение перед удалением. Это показано на рис.3.

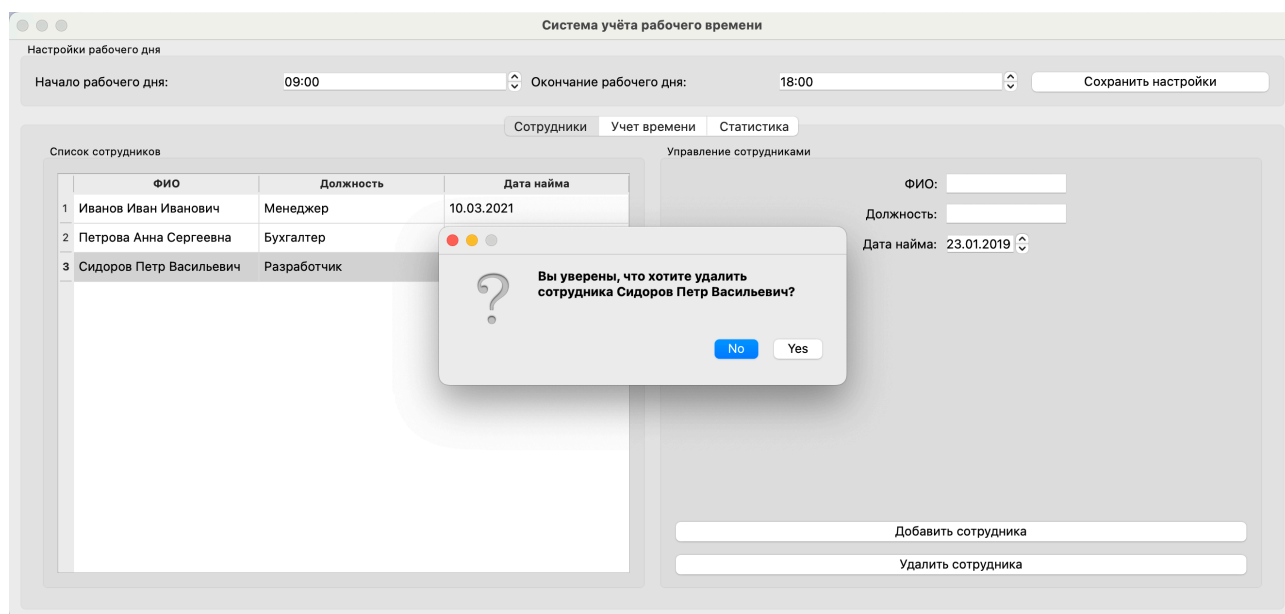


Рис.3 Запрос подтверждения удаления сотрудника

4.2 Настройка рабочего дня

По умолчанию стандартный рабочий день установлен с 9:00 до 18:00. При необходимости эти значения можно изменить с помощью элементов

управления в верхней части окна. На рис.4 показан процесс изменения времени рабочего дня.

Система учёта рабочего времени

Настройки рабочего дня

Начало рабочего дня: 10:00 Окончание рабочего дня: 19:00 Сохранить настройки

Сотрудники Учет времени Статистика

Список сотрудников

	ФИО	Должность	Дата найма
1	Иванов Иван Иванович	Менеджер	10.03.2021
2	Петрова Анна Сергеевна	Бухгалтер	11.05.2023

Управление сотрудниками

ФИО: Должность: Дата найма: 23.01.2019

Добавить сотрудника Удалить сотрудника

Рис.4 Изменение параметров рабочего дня на 10:00 - 19:00

После нажатия кнопки "Сохранить настройки" новые значения сохраняются в базе данных и применяются ко всем расчетам статистики.

4.3 Учет рабочего времени

Для фиксации времени прихода и ухода сотрудников используется вкладка "Учет времени". Сначала выбираем сотрудника из выпадающего списка, затем устанавливаем дату, время прихода и время ухода. Нажимаем кнопку "Сохранить запись". На рис.5 показан этот процесс.

Система учёта рабочего времени

Настройки рабочего дня

Начало рабочего дня: 10:00 Окончание рабочего дня: 19:00 Сохранить настройки

Сотрудники Учет времени Статистика

Выберите сотрудника: Иванов Иван Иванович

Запись времени

Дата: 09.05.2025

Время прихода: 10:15

Время ухода: 19:30

Сохранить запись

История посещений

	Дата	Время прихода	Время ухода
1	09.05.2025	10:15	19:30

Рис.5 Вкладка "Учет времени" при добавлении записи для сотрудника Иванов И.И.

После сохранения записи она появляется в таблице истории посещений. Если на эту дату уже существовала запись, она обновляется. На рис.6 показана таблица после добавления нескольких записей.

Система учёта рабочего времени

Настройки рабочего дня

Начало рабочего дня: 10:00 Окончание рабочего дня: 19:00 Сохранить настройки

Сотрудники Учет времени Статистика

Выберите сотрудника: Иванов Иван Иванович

Запись времени

Дата: 10.05.2025

Время прихода: 10:25

Время ухода: 19:38

Сохранить запись

История посещений

	Дата	Время прихода	Время ухода
1	10.05.2025	10:25	19:38
2	09.05.2025	10:15	19:30

Рис.6 Таблица истории посещений с несколькими записями

Точно также можно вносить время прихода и ухода для других сотрудников. Это показано на рис.7.

Система учёта рабочего времени

Настройки рабочего дня

Начало рабочего дня: 10:00 Окончание рабочего дня: 19:00 Сохранить настройки

Сотрудники Учет времени Статистика

Выберите сотрудника: Петрова Анна Сергеевна

Запись времени

Дата: 09.05.2025

Время прихода: 10:15

Время ухода: 18:52

Сохранить запись

История посещений

	Дата	Время прихода	Время ухода
1	10.05.2025	10:05	19:00
2	09.05.2025	10:15	18:52

Рис.7 Вкладка "Учет времени" при добавлении записи для сотрудника Петрова А.С.

При выборе другого сотрудника в выпадающем списке таблица автоматически обновляется, показывая историю посещений выбранного сотрудника.

4.4 Просмотр статистики

Вкладка "Статистика" позволяет просматривать обобщенные данные о рабочем времени сотрудников. Она разделена на две части: статистика по конкретному сотруднику и статистика по всей компании.

На рис.8 показана статистика для сотрудника Иванов И.И.

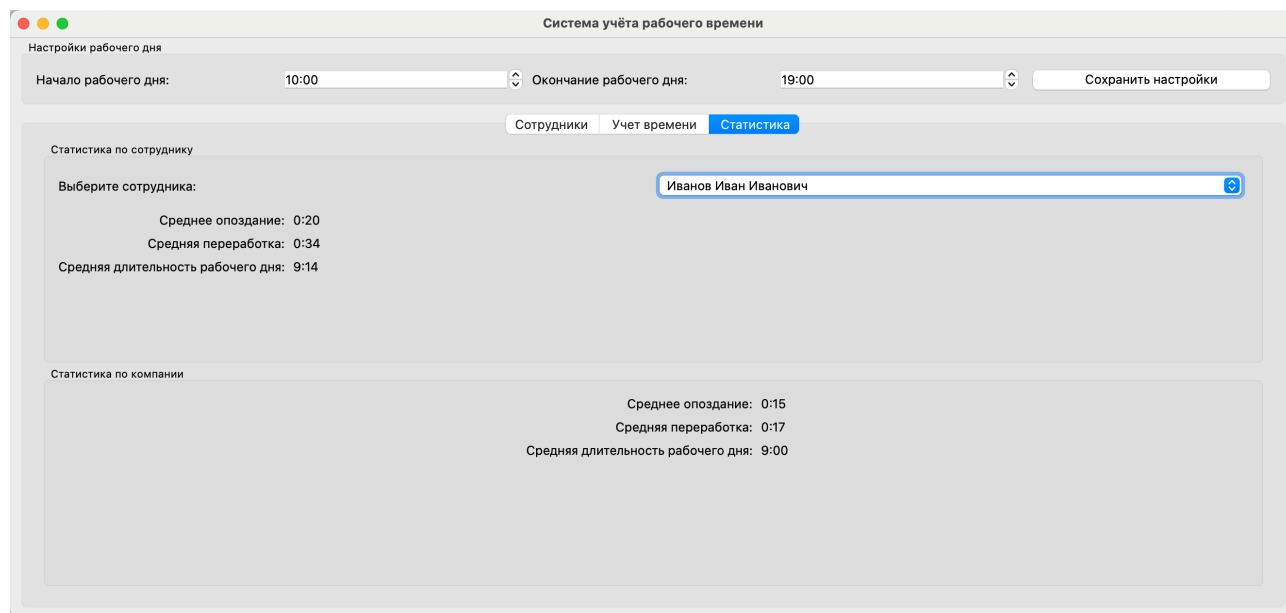


Рис.8 Вкладка "Статистика"

При выборе другого сотрудника в выпадающем списке статистика автоматически пересчитывается. Нижняя часть вкладки отображает средние значения по всем сотрудникам компании.

Из данных на рис.6 и рис.7 видно, что все статистики посчитаны верно.

5. Заключение

Разработанная система учета рабочего времени сотрудников представляет собой эффективный инструмент для автоматизации процессов контроля трудовой дисциплины. Благодаря простому и интуитивно понятному интерфейсу система не требует специальной подготовки пользователей.

Ключевыми преимуществами системы являются:

- Автоматический расчет статистических показателей
- Хранение всех данных в базе данных
- Удобный графический интерфейс
- Модульная архитектура, обеспечивающая возможность дальнейшего расширения функциональности

Система может быть дополнена в будущем такими функциями, как:

- Экспорт данных в различные форматы (Excel, PDF)
- Построение графиков и диаграмм для наглядного представления статистики
- Интеграция с системами контроля доступа для автоматической фиксации времени
- Расширенные отчеты о рабочем времени с учетом выходных и праздничных дней

Таким образом, разработанное приложение не только решает текущие задачи по учету рабочего времени, но и создает основу для построения более сложной и функциональной системы управления персоналом.