

УДК 004.45

РАЗРАБОТКА СЕРВИСА УЧЁТА И КОНТРОЛЯ РАБОЧЕГО ВРЕМЕНИ**Тимофеев Данил Николаевич¹**, бакалаврe-mail: timofeevdanil28@gmail.ru**Синичкина Милана Дмитриевна¹**, бакалаврe-mail: plmrv2000@gmail.ru**Александров Андрей Юрьевич¹**, старший преподаватель кафедры ПМИe-mail: Aalex1986@mail.ru

¹ Казанский национальный исследовательский технический университет имени А.Н. Туполева, Казань, Россия

Аннотация. В работе рассматривается проблема автоматизации учёта рабочего времени в малых и средних коллективах. Предлагается веб-сервис на базе FastAPI и Telegram-бота, позволяющий сотрудникам фиксировать время с помощью таймера, а руководителям — формировать отчёты. Описана архитектура системы, модуль экспорта данных в форматы CSV, XLSX и PDF, а также механизм авторизации через одноразовые коды в Telegram. Результаты работы могут быть использованы для повышения прозрачности учёта трудозатрат.

Ключевые слова: учёт рабочего времени, FastAPI, Python, Telegram-бот, SQLAlchemy, экспорт данных, таймер задач, веб-сервис.

DEVELOPMENT OF A SERVICE FOR ACCOUNTING AND CONTROL OF WORKING HOURS**Danil N. Timofeev¹**, Bachelor's Degree Studente-mail: timofeevdanil28@gmail.ru**Milana D. Sinichkina¹**, Bachelor's Degree Studente-mail: plmrv2000@gmail.ru**Andrey Yu. Alexandrov¹**, Senior lecturer at the Department of PMIe-mail: Aalex1986@mail.ru

¹ Kazan National Research Technical University named after A. N. Tupolev, Kazan, Russia

Abstract. The paper considers the problem of automation of working time accounting in small and medium-sized teams. A web service based on FastAPI and a Telegram bot is offered, which allows employees to record time using a timer, and managers to generate reports. The architecture of the system, the module for exporting data to CSV, XLSX and PDF formats, as well as the authorization mechanism via one-time codes in Telegram are described. The results of the work can be used to increase the transparency of labor accounting.

Keywords: time tracking, FastAPI, Python, Telegram bot, SQLAlchemy, data export, task timer, web service.

Введение

В современных организациях задачи учёта и контроля рабочего времени приобретают особую значимость в условиях цифровой трансформации бизнес-процессов, роста удалённой занятости и необходимости оперативного управления проектами. Точность фиксации временных интервалов, их привязка к конкретным задачам и проектам, а также возможность анализа загрузки сотрудников становятся критическими факторами для принятия управленческих решений и корректного расчёта заработной платы. Однако на практике, особенно в сегменте малого и среднего бизнеса, до сих пор широко распространены ручные, часто анахроничные методы учёта: сотрудники ведут Excel-таблицы с произвольной структурой, делают заметки в корпоративных мессенджерах (Telegram, Slack) или заполняют бумажные журналы и табели. Такой подход неизбежно приводит к целому ряду системных проблем: человеческим ошибкам при вводе данных, неполноте или дублированию записей, значительным задержкам между фактическим выполнением работы и её регистрацией, а также к отсутствию единой централизованной базы для анализа и контроля. Руководители вынуждены вручную агрегировать разрозненную информацию, что отнимает время и снижает эффективность управления.

Существующие на рынке автоматизированные системы для учёта рабочего времени (такие как Toggl Track, Clockify, TimeDoctor, модули Битрикс24) предлагают богатый функционал, однако их внедрение в малых и средних организациях часто сопряжено с трудностями. Во-первых, многие из этих решений имеют платную подписку, что создаёт дополнительную финансовую нагрузку. Во-вторых, они либо перегружены избыточными возможностями (например, скриншотами экрана, геолокацией, интеграциями с десятками сервисов), либо, напротив, не предоставляют гибких механизмов разграничения прав доступа — простой и понятной ролевой модели «руководитель команды – сотрудник». Кроме того, большинство зарубежных сервисов не имеют встроенной поддержки российских мессенджеров и не позволяют выполнять авторизацию и получать уведомления через Telegram — инструмент, который сегодня активно используется для внутренней коммуникации во многих компаниях. Также отсутствует возможность экспорта детализированных отчётов в форматы, удобные для дальнейшего анализа (XLSX, PDF) или импорта в бухгалтерские системы (1C, SAP). Всё это делает существующие решения либо слишком дорогими, либо технически неудобными для массового применения в малых коллективах.

Таким образом, актуальной научно-практической задачей является создание собственного легковесного, но функционального сервиса учёта рабочего времени, который сочетал бы в себе следующие ключевые характеристики: удобную и безопасную авторизацию через Telegram-бота с использованием одноразовых кодов, простой таймер задач с поддержкой пауз и привязкой к проектам, гибкую ролевую модель (администратор, руководитель команды, сотрудник), а также встроенные инструменты для формирования детализированных индивидуальных и командных отчётов с возможностью экспорта в форматах CSV, XLSX и PDF. Целью данной работы является разработка и программная реализация такого сервиса, ориентированного на нужды малых и средних организаций, не требующего значительных затрат на внедрение и обслуживание. Предлагаемое решение призвано устранить недостатки ручных методов учёта и преодолеть ограничения коммерческих систем, обеспечив прозрачность, достоверность и оперативность контроля рабочего времени, а также снизить административную нагрузку на руководителей и сотрудников.



Материалы и методы

Серверная часть сервиса реализована на языке Python 3.12 [4] с использованием асинхронного веб-фреймворка FastAPI [1]. Для доступа к реляционной базе данных применяется SQLAlchemy 2.x (async) [2] с поддержкой PostgreSQL и SQLite. Управление миграциями осуществляется через Alembic. Взаимодействие с пользователями реализовано через Telegram Bot API (библиотека httpx) для отправки одноразовых кодов авторизации и приглашений в команды. Веб-интерфейс построен на статических HTML/CSS/JS с динамическими вызовами Fetch API. Экспорт отчётов обеспечивается библиотеками openpyxl (XLSX) и fpdf2 (PDF). Вся система может быть развёрнута с помощью Docker Compose.

Ключевые сущности базы данных представлены на ER-диаграмме (Рисунок 1). Проектирование реляционной схемы выполнено с учётом принципов, изложенных в [3]. Users (пользователи), Teams (команды), TimeEntries (сессии таймера), TeamInvitations (приглашения). Роли пользователей: сотрудник (employee), руководитель команды (boss), администратор (admin). Механизм учёта времени позволяет запускать, ставить на паузу, возобновлять и останавливать таймер; длительность сессии рассчитывается как разность между временем старта и окончания за вычетом накопленных перерывов (break_seconds).

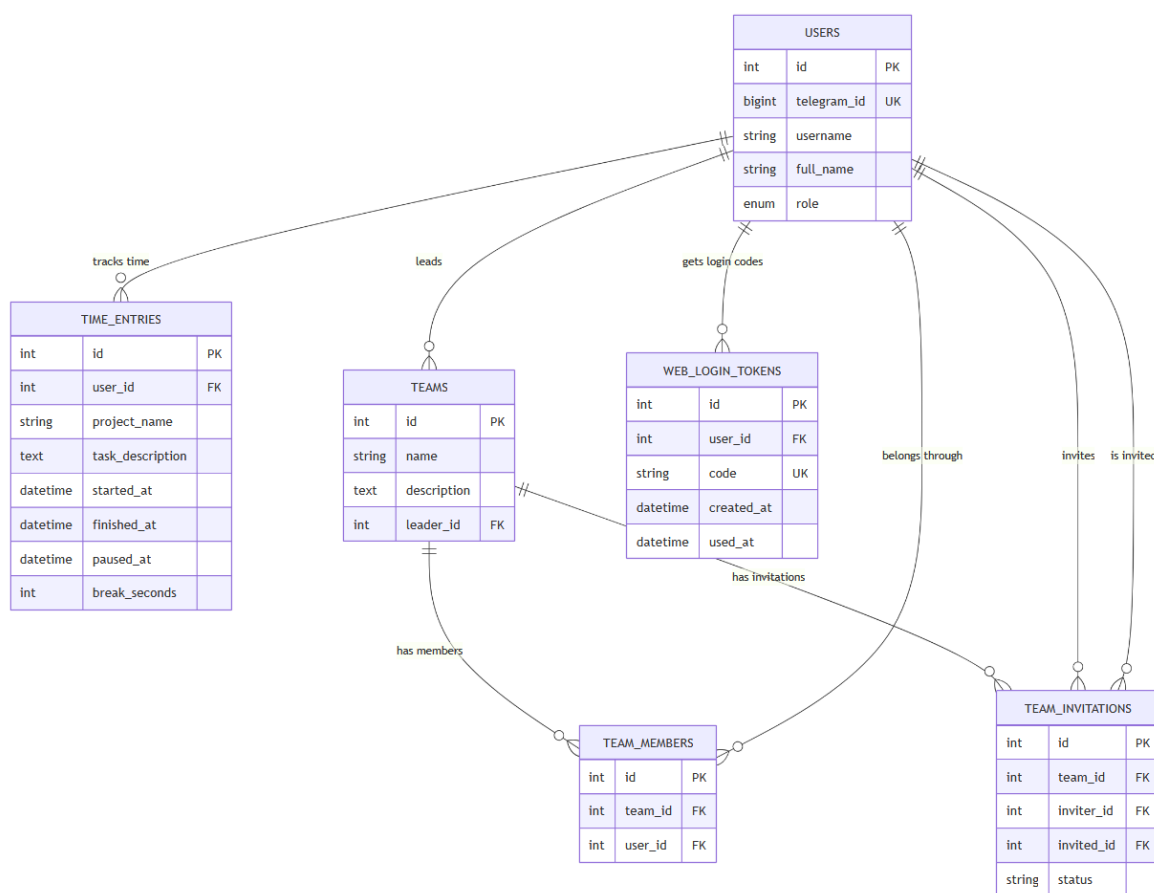


Рисунок 1 — ER-диаграмма базы данных

Результаты

Разработанный сервис (назван «TimeTracker») обеспечивает:

1. Авторизацию без пароля: пользователь вводит свой Telegram ID или @username на веб-странице, система отправляет одноразовый код в Telegram, после ввода кода пользователь получает доступ к личному кабинету.

2. Учёт времени через таймер: сотрудник выбирает проект и задачу (или оставляет пустыми), нажимает «Старт». Таймер можно поставить на паузу и возобновить. Активная сессия всегда отображается в верхней панели сайта.

3. Ролевое разграничение: руководитель команды видит статистику всех участников своей команды (сегодня, за неделю, за период), может приглашать и удалять участников через Telegram-бота.

4. Экспорт отчётов: индивидуальная и командная статистика может быть выгружена в форматах CSV, XLSX и PDF с меткой времени формирования в локальном часовом поясе пользователя.

На рисунке 2 показана схема авторизации через Telegram-бота.

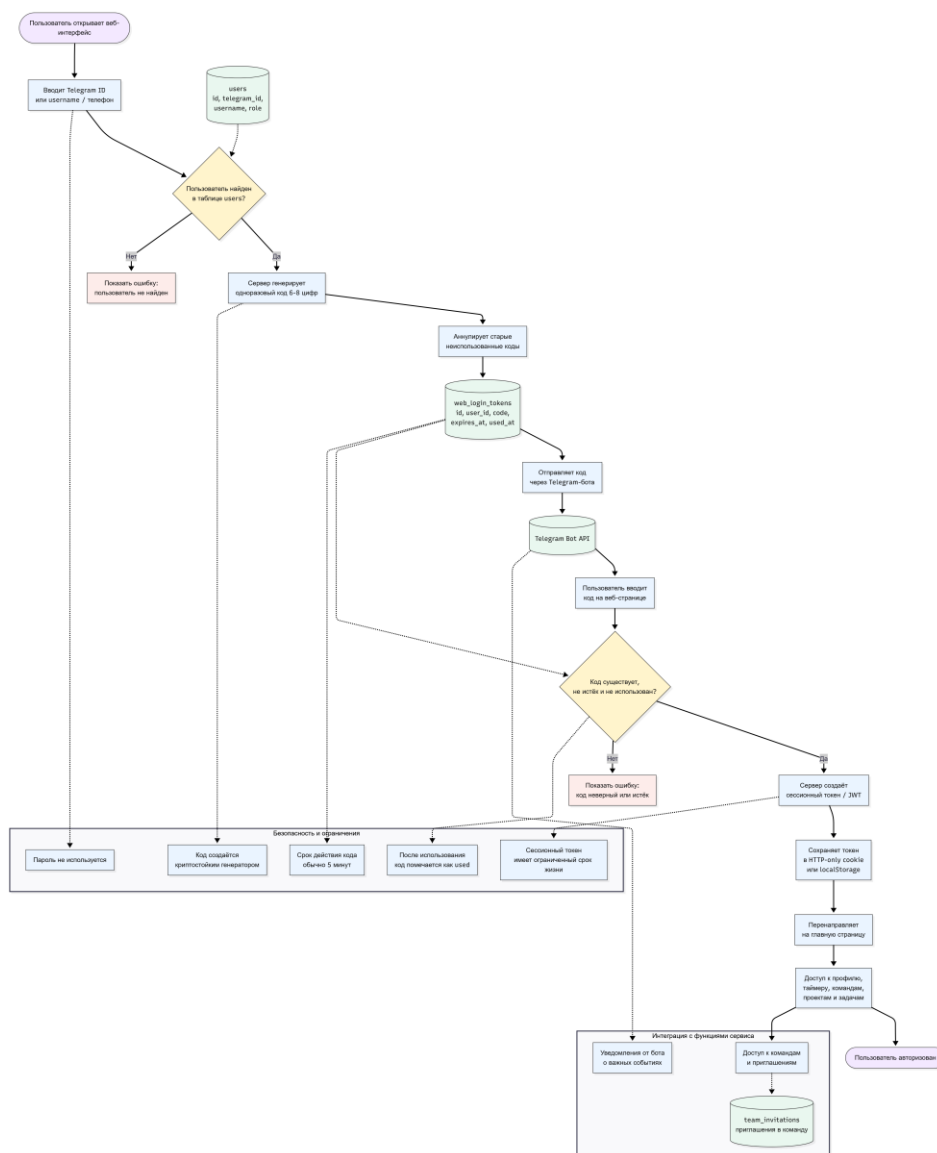


Рисунок 2 — Схема процесса авторизации

Заключение

В ходе выполнения данной работы успешно разработан и протестирован сервис учёта и контроля рабочего времени «TimeTracker». Все поставленные цели и задачи достигнуты в полном объёме. В частности, реализован полнофункциональный таймер задач с поддержкой операций запуска, паузы, возобновления и остановки, а также привязкой

временных сессий к проектам и задачам. Выполнена глубокая интеграция с Telegram-ботом: пользователи могут авторизоваться в системе без пароля, используя одноразовые коды, получать приглашения в команды и при необходимости — уведомления о событиях. Реализована гибкая ролевая модель «руководитель команды – сотрудник – администратор», обеспечивающая разграничение прав доступа к личной и командной статистике. Сервис позволяет формировать детализированные индивидуальные и командные отчёты за день, неделю или произвольный период с возможностью экспорта результатов в форматах CSV, XLSX и PDF, что удобно для дальнейшего анализа и интеграции с бухгалтерскими системами.

Разработанный сервис готов к практическому внедрению в малых и средних организациях. Он не требует дорогостоящего коммерческого ПО, легко развёртывается (в том числе через Docker Compose) и прост в освоении для конечных пользователей. Основные перспективы дальнейшего развития включают: добавление функции ручного редактирования временных сессий для руководителей и администраторов (для исправления ошибок), интеграцию с распространёнными бухгалтерскими системами (например, 1С), реализацию фоновых задач (автоматические напоминания о незавершённом таймере, еженедельная рассылка отчётов), а также разработку прогрессивного веб-приложения (PWA) или мобильного приложения для удобной фиксации времени вне офиса.

Список литературы:

1. FastAPI Documentation. – URL: <https://fastapi.tiangolo.com> (дата обращения: 01.05.2026)
2. SQLAlchemy Documentation. – URL: <https://docs.sqlalchemy.org/en/20> (дата обращения: 01.05.2026)
3. Элмасри Р., Навате Б. Основы систем баз данных. – СПб.: Питер, 2008. – 992 с.
4. Любанович Б. Python. К вершинам мастерства. – М.: ДМК Пресс, 2023. – 960 с.

