# TaskPulse — Backend Design Specification

## 1. Назначение и область документа

Документ описывает серверную часть (Backend) системы TaskPulse: доменные модели, бизнес-функции, роли и права, интеграции, фоновые задачи и нефункциональные требования. Документ служит источником истины для разработчиков, QA и DevOps. Фронтенд и UX вне рамок этого документа.

## 2. Технологический стек и архитектура

• Язык и фреймворки: Python 3.13, Django 5.x, Django REST Framework.  
• База данных: PostgreSQL.  
• Веб-сервер/прокси: Nginx.  
• Интеграции: Telegram Bot API (webhook), SMTP (email).

## 3. Обзор доменной модели

Система управляет задачами (Task), пользователями (User), вложениями (Attachment), комментариями (Comment), отправленными уведомлениями (NotificationLog) и отчётами (Report). Основные роли: CREATOR, EXECUTOR. Создатель назначает задачи исполнителям, система отправляет напоминания за 24 часа до срока через Telegram.

## 4. Модели данных (Django ORM)

### 4.1 User

Назначение: хранение учётных записей, ролей и идентификаторов интеграций. Пользователь может быть создателем и/или исполнителем задач.

Таблица полей модели «User»:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поле | Тип | Обяз. | Ограничения/по умолчанию | Описание |
| id | UUID | да | PK, генерируется | Идентификатор пользователя |
| email | string(255) | да | UNIQUE, lower-case | Email для логина и уведомлений |
| password\_hash | string | да | bcrypt/argon2 | Хэш пароля |
| role | enum[CREATOR, EXECUTOR, ADMIN] | да | DEFAULT=EXECUTOR | Глобальная роль |
| first\_name/last\_name | string(120) | нет | - | Имя/фамилия |
| timezone | string | нет | DEFAULT=UTC | Часовой пояс пользователя |
| telegram\_id | string | нет | UNIQUE, nullable | Связка с аккаунтом Telegram |
| is\_active | bool | да | DEFAULT=true | Признак активной учётной записи |
| created\_at/updated\_at | timestamp | да | auto | Системные поля аудита |

Индексы: UNIQUE(email), INDEX(telegram\_id).

Бизнес-правила: email в нижнем регистре, подтверждение email, блокировка при множественных неудачных входах.

### 4.2 Task

Назначение: основная сущность для управления работой. Содержит ключевые атрибуты задачи, срок выполнения, приоритет и статусы жизненного цикла.

Таблица полей модели «Task»:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поле | Тип | Обяз. | Ограничения/по умолчанию | Описание |
| id | UUID | да | PK | Идентификатор задачи |
| title | string(255) | да | - | Краткое название |
| description | text | нет | - | Подробное описание |
| creator\_id | UUID→User | да | FK, INDEX | Создатель |
| assignee\_id | UUID→User | нет | FK, INDEX, nullable | Исполнитель |
| priority | enum[LOW, MEDIUM, HIGH] | да | DEFAULT=MEDIUM | Приоритет задачи |
| status | enum[OPEN, IN\_PROGRESS, DONE, OVERDUE] | да | DEFAULT=OPEN | Текущий статус |
| due\_at | timestamp | нет | INDEX, nullable | Дедлайн |
| closed\_at | timestamp | нет | nullable | Фактическое завершение |
| postponed\_count | int | да | DEFAULT=0 | Сколько раз переносили срок |
| created\_at/updated\_at | timestamp | да | auto | Системные поля аудита |

Бизнес-правила: только CREATOR/ADMIN могут создавать и назначать; исполнитель может менять статус в рамках допустимых переходов; перенос срока логируется; DONE запрещён редактировать кроме комментариев/вложений.

### 4.3 NotificationLog

Назначение: журнал отправки уведомлений и действия получателей (кнопки в Telegram).

Таблица полей модели «NotificationLog»:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поле | Тип | Обяз. | Ограничения/по умолчанию | Описание |
| id | UUID | да | PK | Идентификатор |
| task\_id | UUID→Task | да | FK, INDEX | Задача |
| recipient\_id | UUID→User | да | FK | Кому отправлено |
| channel | enum[telegram,email] | да | - | Канал доставки |
| type | enum[assign, reminder24h, confirm, postpone] | да | - | Тип уведомления |
| payload | jsonb | нет | nullable | Данные запроса/ответа внешнего сервиса |
| status | enum[queued, sent, delivered, failed] | да | DEFAULT=queued | Статус доставки |
| delivered\_at | timestamp | нет | nullable | Когда доставлено |
| created\_at | timestamp | да | auto | Когда создана запись |

Бизнес-правила: ретраи фиксируются; idempotency по внешним message\_id (если доступно).

## 5. Роли и доступы (RBAC)

• CREATOR: создаёт/редактирует/удаляет свои задачи, назначает исполнителей, переносит сроки.  
• EXECUTOR: видит назначенные задачи, меняет статус в допустимых переходах, добавляет комментарии и вложения.  
• ADMIN: все права, системные операции (пользователи, реиндексация, очистка).

## 6. Основные функции и сервисы

### 6.1 Сервис аутентификации и профиля

Регистрация, вход/выход, обновление токенов, смена/сброс пароля, подтверждение email, обновление профиля (имя, часовой пояс), привязка telegram\_id.

### 6.2 Сервис задач (Task Service)

• Создание задачи (CREATOR/ADMIN), назначение исполнителя.  
• Фильтрация и поиск задач (по статусу, приоритету, срокам, участникам, тексту, сортировке).  
• Переходы статусов: OPEN→IN\_PROGRESS→DONE; REOPEN из DONE; автоматический OVERDUE при просрочке.  
• Перенос срока (postpone) с фиксацией причины и ограничениями по количеству переносов (настраиваемо).  
• Подтверждение «выполню вовремя» со стороны исполнителя; логирование событий.  
• История изменений (аудит): статус, исполнитель, сроки.

### 6.3 Сервис уведомлений (Notification Service)

• Отправка сообщений в Telegram (назначение задачи, напоминания T−24h, подтверждения, переносы).  
• Инлайн-кнопки (callback): «Продлить на сутки», «Сделаю вовремя».  
• Хранение статусов доставки и ретраев в NotificationLog.

## 8. Интеграции

• Telegram Bot API: входящий webhook.  
• Email SMTP: подтверждение email, восстановление пароля, системные уведомления (опционально).