

Тестовое задание: Веб-сервис для управления движением денежных средств (ДДС)

Описание:

ДДС (движение денежных средств) — это процесс учета, управления и анализа поступлений и списаний денежных средств компании или частного лица. В рамках данного задания пользователь должен иметь возможность вести учет всех денежных операций с учетом следующих параметров:

Вам необходимо -разработать веб-приложение, которое позволяет пользователям создавать, редактировать, удалять и просматривать записи о движении денежных средств. Веб-приложение должно быть удобным, иметь интуитивно понятный интерфейс и обеспечивать соблюдение логических зависимостей между сущностями.

Возможности приложения:

- **Создание записи о движении денежных средств (ДДС)::**
 - Поля:
 - **Дата создания записи** — заполняется автоматически, но может быть изменена вручную. Пример записи — 01.01.2025
 - **Статус** — имеет следующие значения:
 - Бизнес
 - Личное
 - Налог
 - *Данный список должен иметь возможность расширяться*
 - **Тип** — имеет следующие значения:
 - Пополнение
 - Списание
 - *Данный список должен иметь возможность расширяться*
 - **Категория и подкатегория** — пример значений:
 - Категория “Инфраструктура” (подкатегории: "VPS", "Proxy")
 - Категория “Маркетинг” (подкатегории: "Farpost", "Avito")
 - *Данный список должен иметь возможность расширяться*
 - **Сумма** — количество средств в рублях, например, 1 000 р.:
 - **Комментарий** — комментарий к записи в свободной форме (должен быть необязательным к заполнению).:
- **Просмотр списка всех записей:**
 - Вывод таблицы с данными: дата, статус, тип, категория, подкатегория, сумма, комментарий.
 - Поддержка фильтрации по дате (с указанием периода дат), статусу, типу, категории и подкатегории.

- **Редактирование записи::**
 - Возможность изменить любую запись.
- **Удаление записи::**
 - Возможность удалить любую запись.
- **Управление справочниками:**
 - Добавление, редактирование и удаление статусов, типов, категорий и подкатегорий.
- **Логические зависимости:**
 - Подкатегории должны быть привязаны к категориям.
 - Категории должны быть привязаны к типам. Например, в тип “Списание” относится категория “Маркетинг”, в которую входят подкатегории "Farpost", "Avito".

Требования к реализации:

- **Технологии::**
 - Python: Django (с Django ORM)
 - База данных: SQLite (для упрощения) или любая другая реляционная база данных.
 - Frontend: для создания frontend-части данного веб-приложения можно воспользоваться стандартной админ-панелью джанго или разработать собственный интерфейс с использованием HTML, CSS (Bootstrap), JavaScript (по необходимости).
- **Основные страницы веб-приложения::**
 - Главная страница:
 - Таблица с записями о ДДС.
 - Фильтры по дате, статусу, типу, категории и подкатегории.
 - Страница создания/редактирования записи:
 - Форма для ввода данных.
 - Поле выбора категорий, автоматически фильтрующее подкатегории на основе выбранной категории.
 - Страница управления справочниками:
 - Добавление, редактирование и удаление статусов, типов, категорий и подкатегорий, а также установление необходимых зависимостей.
- **Бизнес-правила:**
 - Пользователь не может выбрать подкатегорию, если она не связана с выбранной категорией.

- Пользователь не может выбрать категорию, если она не относится к выбранному типу.
- **Валидация данных::**
 - Поля "сумма", "тип", "категория" и "подкатегория" обязательны.
 - Валидация на стороне клиента и сервера.

Ожидаемый результат:

1. Рабочее веб-приложение с удобным интерфейсом.
2. Инструкция по запуску проекта:
 - Как установить зависимости.
 - Как настроить базу данных.
 - Как запустить веб-сервис.
3. Исходный код должен быть загружен в GitHub-репозиторий.
4. (Опционально) Скриншоты интерфейса или ссылка на демо-версию.

Оценочные критерии:

1. **Код:**
 - Чистота и структура кода.
 - Корректное использование Django ORM и Django REST Framework.
 - Наличие адекватных комментариев и документации.
2. **Функциональность:**
 - Полное выполнение всех требований задания.
 - Корректность работы всех функций веб-сервиса.
3. **Интерфейс:**
 - Минимальный пользовательский интерфейс.
 - Качество UX: интуитивно понятное управление записями и справочниками.
4. **Документация:**
 - Понятная инструкция по запуску.

Срок выполнения задания: 5 дней с момента получения тестового задания.