

создано с
росмолодёжь
гранты



**20
26**

ОТДЕЛ ЛОГИСТИКИ И ИТ-ОБЕСПЕЧЕНИЯ

IT-специалист

Дедлайн отправки тестового задания:

2 октября, 23:59

Загрузи файлы своего решения через личный кабинет в нашем Telegram-боте:

@cbc_team_bot





АРТЁМ ЗОБКОВ

Руководитель отдела логистики и IT-обеспечения

你好! Спасибо, что подал(-а) заявку в отдел логистики и IT-обеспечения КБК'26! Наш отдел запускает механизм работы КБК и следит, чтобы каждая деталь была на своём месте: именно мы организуем закупки, ведём бюджет и выстраиваем логистику, координируем волонтеров и обеспечиваем безупречную техподдержку.

Сейчас тебе предстоит выполнить небольшое тестовое задание, которое поможет оценить твои навыки и сильные стороны и покажут твою готовность к работе в отделе.

Желаю удачи!

ТЗ IT-СПЕЦИАЛИСТА

Задание состоит из двух частей:

1. Верстка сайта;
2. Создание бота.

От кандидата ожидается **один PDF-файл** с:

1. Ссылкой на репозиторий сайта;
2. Ссылка на сайт GitHub Pages;
3. Ссылка на репозиторий бота;
4. *Опционально:* комментарии по решению.

ЗАДАНИЕ 1

Верстка сайта

Нужно создать простой сайт-визитку. Над дизайном парится не надо. В первую очередь смотрим на верстку. Код проекта должен лежать в публичном репозитории GitHub. Используй Git с самого начала, чтобы мы видели весь процесс разработки.

Структура (HTML)

Главная страница должна содержать:

- логотип (любой – можно заглушку);
- меню с минимум 3 ссылками (например: «О компании», «Расписание», «Контакты»);
- блок «О компании» с текстом и картинкой;
- Блок частые вопросы (показываются вопросы и по нажатию выпадают ответы — пример forum-cbc.com);
- кнопка "Зарегистрироваться" и всплывающая форма регистрации (имя, email, сообщение);
- футер с контактами.

Стилизация (CSS)

Использовать Flexbox или CSS Grid, либо что-то иное. Главное — адаптивность для всех устройств.

Обеспечить адаптивность:

- на экране шириной $> 1024\text{px}$ меню горизонтальное;
- на $768\text{--}1024\text{px}$ меню сжимается;
- $< 768\text{px}$ меню превращается в «бургер» (JS).

Интерактивность (JS)

- Реализовать раскрывающееся меню-бургер (см. выше);
- Проверка полей формы: нельзя отправить пустое сообщение, email должен быть в формате example@mail.com;
- При успешной отправке формы — выводится сообщение «Спасибо, мы свяжемся с вами» (можно без реальной отправки, просто имитация).

Дополнительно

- Все картинки должны быть оптимизированы (не более 200kb).
- У каждой картинки есть alt.
- Указать title и meta description на странице.
- Подключить favicon.

Деплой

- Код проекта должен лежать в публичном репозитории GitHub. Используйте его с самого начала, чтобы мы видели весь процесс разработки.
- Сайт должен быть выложен на GitHub Pages. Справка: <https://docs.github.com/en/pages/getting-started-with-github-pages/creating-a-github-pages-site>

ЗАДАНИЕ 2. БОТ

Надо написать бота. В этом задании предусмотрено два уровня сложности. Первый – общий, второй – “со звездочкой”.

На втором уровне надо настроить интеграции с другими технологиями и сервисами. Его делать не обязательно.

Базовый уровень

Требования

- aiogram ($\geq 3.11.0$) + aiogram_dialog ($\geq 2.2.0$)
- Конфиг токена бота через .env

Пример: https://drive.google.com/file/d/1qk0GVnpHiQcvhtVQNs3dW_6yUJMnfui-/view?usp=sharing

Структура бота:

(Дана в контексте терминов `aiogram_dialog` – диалог, окно, виджет).

Хендлер на команду `/start` → Диалог Главное меню

Диалог 1 – Главное меню

Два окна:

Главное меню

Виджет фото (любое)

Кнопка 1: Регистрация → Диалог Регистрация

Кнопка 2: Поддержка → Окно Поддержка

Поддержка (с контактом поддержки)

Кнопка 1: Назад → Окно Главное меню

Диалог 2 – Регистрация

Три окна:

Информация о регистрации

Кнопка 1: Далее → Окно Запрос ФУО

Кнопка 2: Отмена → Диалог Главное меню

Запрос ФУО

Кнопка 1: Далее → Окно Подтверждение

Кнопка 2: Отмена → Диалог Главное меню

Подтверждение

В сообщении выводим введенное ФУО

Кнопка 1: Подтвердить и отправить → Отправить в чат введенное ФУО → Диалог Главное меню

Кнопка 2: Изменить → Окно Запрос ФУО

Кнопка 3: Отмена → Диалог Главное меню

*Продвинутый уровень

Требования

- aiogram ($\geq 3.11.0$) + aiogram_dialog ($\geq 2.2.0$)
- postgres (async)
- redis
- *Google sheets API
- docker (опционально)
- Конфиг токена бота через .env

Пример: https://drive.google.com/file/d/1qk0GVnpHiQcvhtVQNs3dW_6yUJMnfui-/view?usp=drive_link

Структура бота

Структура остается та же.

Задание

- Настройка redis как storage диспетчера
- Сохранение ФУО в БД postgres (Используем модель БД. Можно свою написать, можно через ORM, – например, SQLAlchemy)
- Таблица users; PK по tg_id. Сохраняем юзернейм и ФУО (*опционально: разделить ФУО на 3 столбца)
- Создание документации для настройки бота
- Создание .env.example
- *ЗАДАНИЕ ПОД ВТОРОЙ ЗВЕЗДОЧКОЙ Сохранение информации из БД в Гугл Таблицу