# **Техническое задание (ТЗ) на разработку медицинского административного портала**

## **Общее описание проекта**

### **Назначение**

Разработка современного административного веб-приложения для медицинского учреждения с основными функциями:

* Панель управления пациентами и врачами
* Запись на прием
* Мониторинг медицинской статистики
* Управление медицинскими данными и расписанием
* Интерактивные дашборды и отчеты

## **Технический стек**

* **Backend**: .NET WebAPP, EF Core
* **База данных**: PostgreSQL

## **Структура сайта**

### **Общий макет**

#### **Header:**

* Логотип учреждения
* Иконки уведомлений, сообщений и профиля
* Кнопка быстрого поиска

#### **Sidebar:**

* Навигационное меню (с выпадающими пунктами):  
  + Dashboard
  + Пациенты
  + Врачи
  + Расписание
  + Статистика
  + Заявки
  + Настройки

#### **Контентная область:**

* Меняется динамически в зависимости от выбранного пункта меню

## **Основные страницы**

### **Dashboard**

#### **Назначение:**

Предоставление обзорной информации о текущем состоянии медицинского учреждения

#### **Элементы:**

* Статистика: кол-во пациентов, врачей, записей
* Графики: посещаемость, загрузка врачей, категории обращений
* Список последних заявок

### **Пациенты (/patients)**

#### **Назначение:**

Управление и просмотр информации о пациентах

#### **Структура:**

* Таблица пациентов:  
  + ФИО
  + Возраст
  + Статус
  + Контактные данные
* Фильтры и поиск
* Кнопка "Добавить пациента"
* Кнопка "Редактировать / Удалить"

### **Врачи (/doctors)**

#### **Назначение:**

Просмотр и управление информацией о врачах

#### **Структура:**

* Карточки врачей
* Информация:  
  + Имя
  + Специализация
* Возможность редактирования и добавления врача

### **Расписание (/appointments)**

#### **Назначение:**

Управление и просмотр графика приемов

#### **Элементы:**

* Календарь с режимами "день / неделя / месяц"
* Поиск по врачу / пациенту
* Модальное окно создания записи:  
  + Врач, пациент, дата, время, описание
* Цветовое кодирование по типу приема

### **Заявки (/requests)**

#### **Назначение:**

Обработка заявок на прием, регистрацию и услуги

#### **Структура:**

* Таблица заявок  
  + ФИО пациента
  + Контактные данные
  + Услуга
  + Дата подачи
* Кнопки: "Принять", "Отклонить", "Перенести"

### **Страница профиля (/profile)**

#### **Структура:**

* Фото пользователя
* Информация: ФИО, должность, email, номер телефона
* Настройки пароля и уведомлений

## **Функциональные требования**

### **Общие**

* Полная адаптивность
* Поддержка темной/светлой темы
* Интерактивные графики и отчеты
* Локализация

### **Записи**

* Создание / редактирование / удаление записей
* Проверка пересечений в расписании

### **Уведомления**

* Email-оповещения пациенту/врачу

## **Email-шаблоны**

### **Письмо пациенту:**

**Тема:** Подтверждение записи на прием  
 **Текст:**

Здравствуйте, [ФИО]!

Ваша запись на прием успешно подтверждена:

- Врач: [ФИО врача]

- Дата и время: [дата и время]

- Услуга: [наименование]

Вы можете изменить или отменить запись через личный кабинет.

С уважением,

Администрация Doclinic

### **Письмо администратору:**

**Тема:** Новая заявка на прием #ID[номер]  
 **Текст:**

Поступила новая заявка:

- ФИО пациента: [ФИО]

- Телефон: [телефон]

- Email: [email]

- Услуга: [наименование]

- Дата и время заявки: [время]

## **Дизайн-требования**

* Использовать шаблон [Medical Admin Template](https://medical-admin-template.multipurposethemes.com/main/index.html) без модификаций ядра
* Цветовая схема: по умолчанию из шаблона
* Стиль карточек, таблиц– согласно шаблону
* Респонсив:  
  + Mobile (320px–768px): свернутое меню, вертикальный контент
  + Tablet (768px–1024px): 2-колоночный режим
  + Desktop (1024px+): полный функционал

### **Деплой:**

* **Frontend**: сборка шаблона → npm run build → Nginx
* **Backend**: API-сервер в Docker-контейнере
* **База данных**: отдельный контейнер PostgreSQL