

**Обязательные технические требования:**

react-router-dom для роутинга

Использование готового API, расположенного в папке server этого репозитория

Исходный код решения должен быть выложен с вашего аккаунта на GitHub с readme-файлом, содержащим инструкцию по запуску проекта и обоснованием выбора необязательных технологий

**Дополнительные технические требования:**

Разрешено использование любой библиотеки UI-компонентов

Желательно использование TypeScript

Разрешено использование любых внешних библиотек, в том числе:

Дизайн-система/UI-kit (Material-UI, Ant Design)

Стейт-менеджмент (Redux, MobX, Effector)

Линтер (ESLint)

Prettier

Система сборки (Webpack, Vite)

Библиотека для работы с асинхронными HTTP-запросами (React Query, Axios)

Возможность запустить проект в контейнеризированной docker-среде, ещё лучше — сервер и клиент запускаются при помощи docker compose

Прерывание (отмена/прекращение) запросов при переходе со страницы на страницу

Покрытие кода Unit-тестами

Комментарии к коду и документация

Функциональные требования

Просмотр всех задач: отображение всех созданных задач

Просмотр досок: отображение всех досок

Просмотр доски: отображение доски с просмотром задач и краткой информации по ним

Детальный просмотр задачи: модальное окно с возможностью просмотра/редактирования детальной информации о задаче

Создание задачи: возможность создать тикет и прикрепить его к нужной доске

Редактирование задачи со страницы всех задач: возможность со страницы всех задач посмотреть/отредактировать детальную информацию о задаче

Редактирование задачи со страницы доски: возможность на странице доски посмотреть/отредактировать детальную информацию о задаче

Возможность со страницы всех задач перейти на страницу доски с открытием детальной информации о выбранной задаче

Header: Есть header при помощи которого можно всегда перейти на страницы:

Список всех задач

Список досок

Кнопка создания задачи

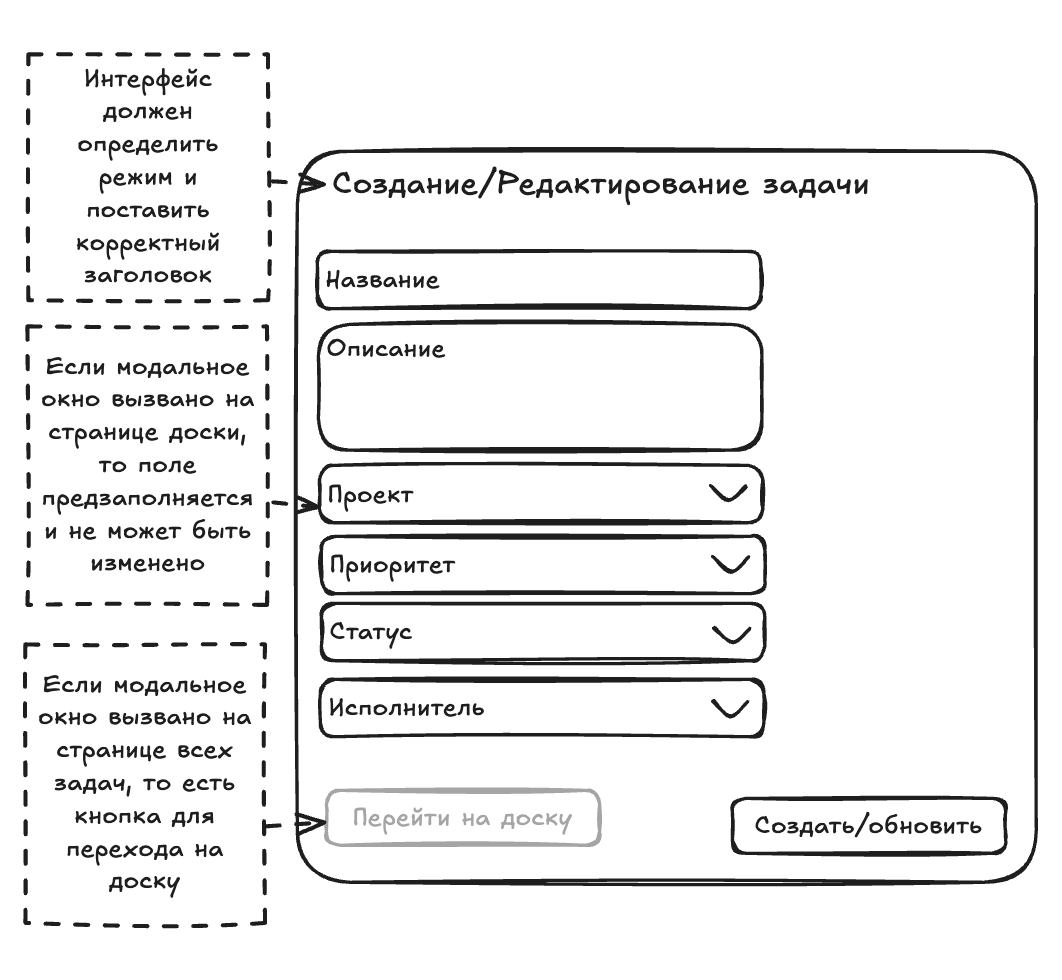
Маршрутизация

/boards — страница, на которой отображены все доски

/board/:id — страница доски проекта

/issues — страница всех задач всех проектов

**Форма создания задачи**

****

Необходимо реализовать форму создания/редактирования задачи. Предполагается, что она будет находиться в модальном окне или drawer-панели. Форма должна поддерживать предзаполненные поля.

**Форма может быть вызвана:**

На любой странице через header

На странице всех задач

На странице доски

Содержание формы:

Название задачи - текстовое поле

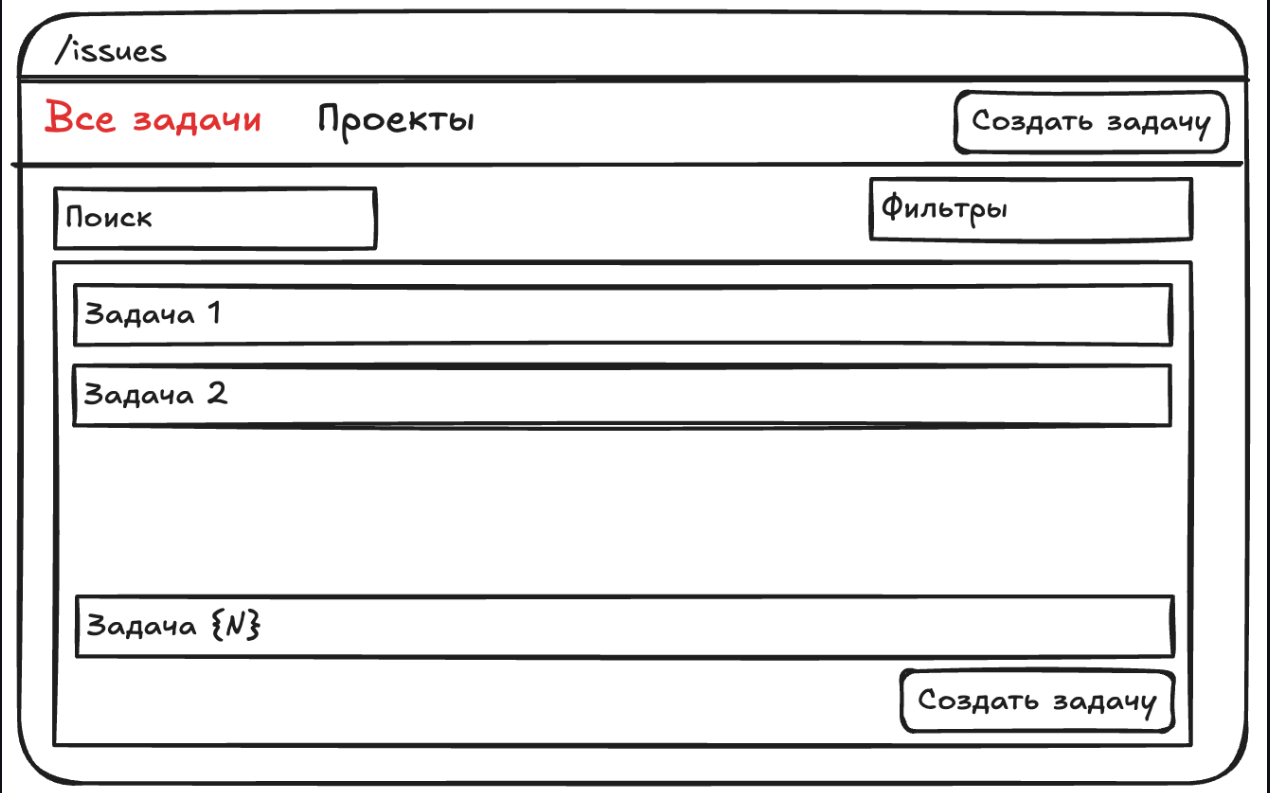
Описание задачи - многострочное текстовое поле

priority - селектор

status - селектор

Исполнитель - селектор

**Страница всех задач**



-На странице есть кнопка «Создать задачу». После создания задачи список должен обновиться

-При клике на задачу открывается modal/drawer с заполненной формой. Все параметры задачи можно обновить. Есть кнопка, кликнув на которую можно перейти на доску, к которой прикреплена задача. Должна открыться страница доски и modal/drawer с предзаполными полями задачи

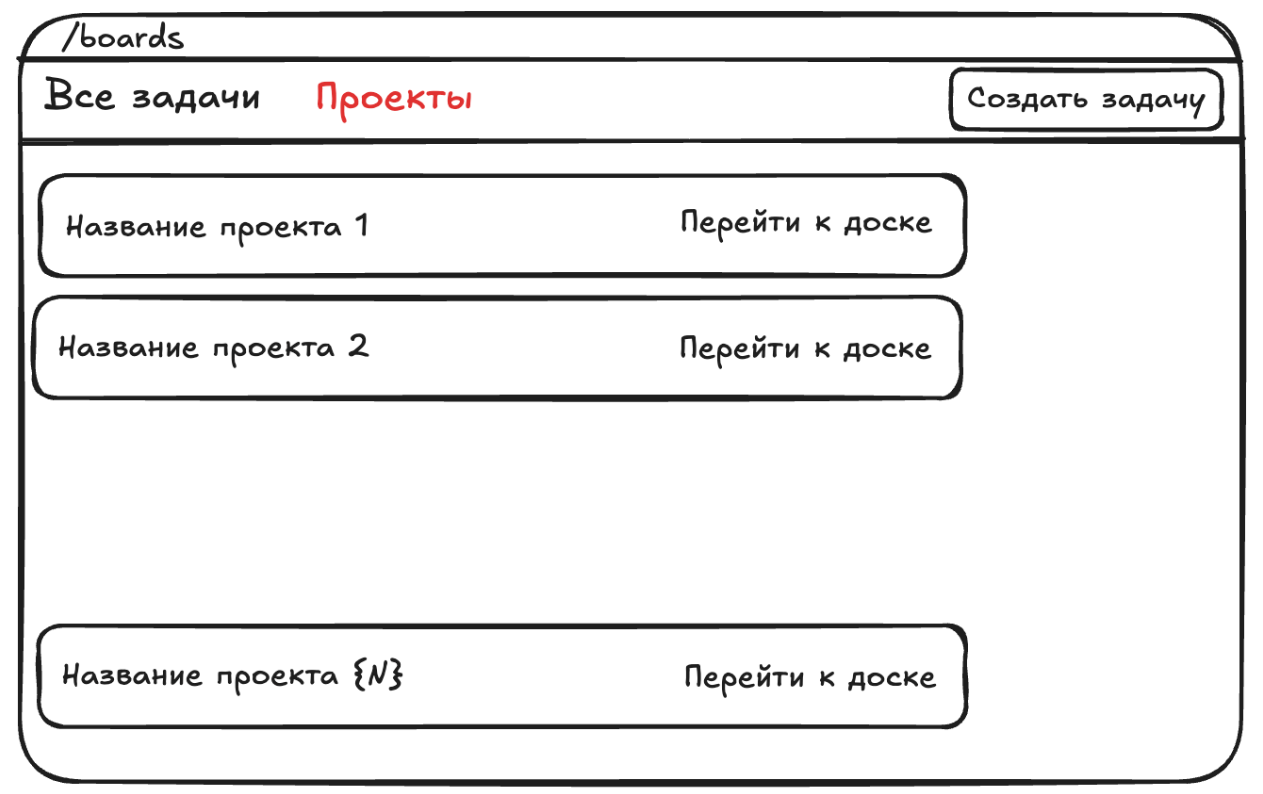
-Фильтр по статусу задачи

-Фильтр по доске, к которой он привязан

-Поиск по названию задачи

-Поиск по исполнителю

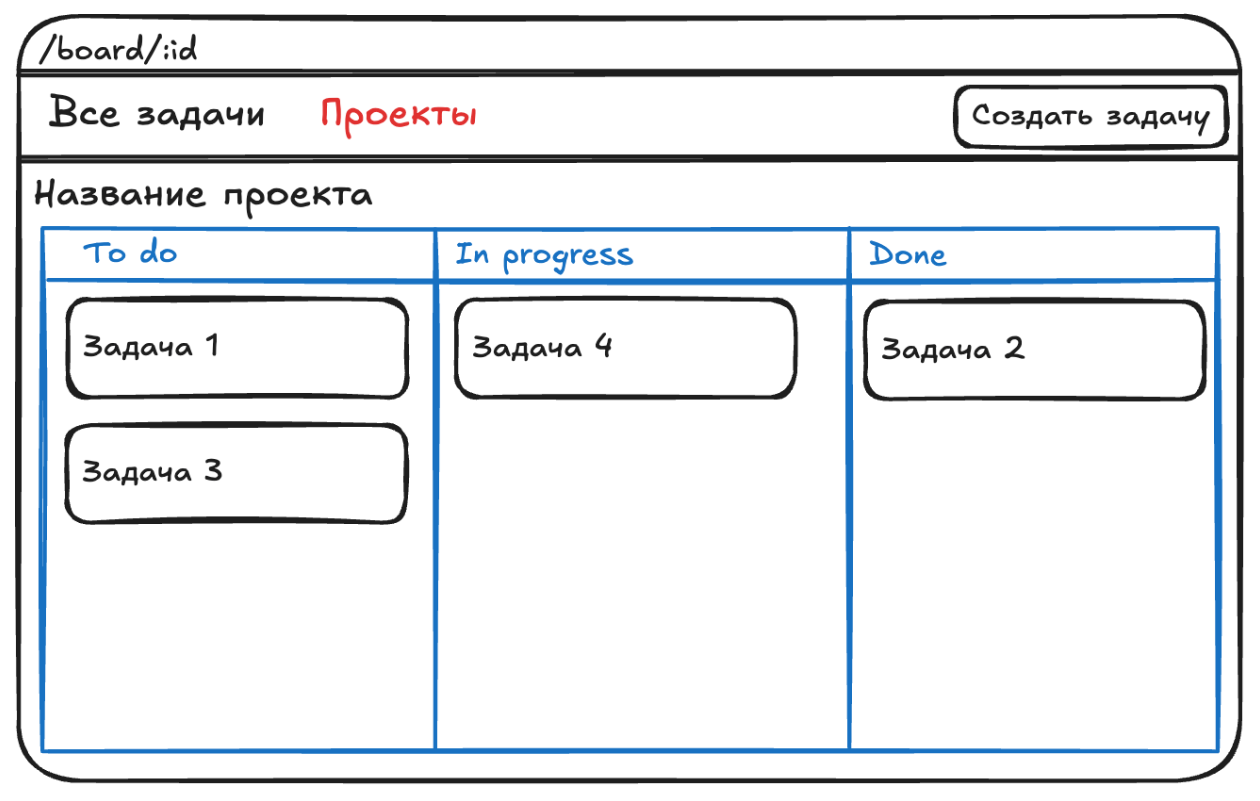
**Страница проектов**



-На странице отображен список всех проектов(досок)

-Из страницы можно перейти на страницу выбранного проекта

**Страница доски**



-На странице доски отображаются все задачи этого проекта

-На любую задачу можно нажать, тем самым вызвав окно редактирования с предзаполненными полями

-Задачи разделены на колонки по их статусам. При изменении статуса задачи, она должна перейти в другую колонку без перезагрузки страницы

**Дополнительно**

Смена статуса задачи на доске посредством Drag-and-drop

При перезагрузке страницы данные формы должны сохраняться в черновик