

Kanban Board Component

A production-quality, fully functional Kanban Board View built with React, TypeScript, and Tailwind CSS. This implementation demonstrates enterprise-grade UI/UX patterns, performance optimization, and accessibility-first approach.

Features

✓ **Drag-and-Drop Functionality**

- Smooth drag-and-drop using @dnd-kit
- Reorder tasks within columns
- Move tasks between columns
- Keyboard navigation support

✓ **Task Management**

- Create, edit, and delete tasks
- Task priority levels (Low, Medium, High, Urgent)
- Assignee management
- Tags and labels
- Due dates with overdue indicators
- Rich task descriptions

✓ **Column Management**

- Multiple status columns (To Do, In Progress, Review, Done)
- WIP (Work In Progress) limits
- Visual progress indicators
- Task count badges

✓ **Advanced Features**

- Fully accessible (WCAG 2.1 AA compliant)
- Responsive design (mobile, tablet, desktop)
- Keyboard navigation
- Modal-based task editing
- Empty state handling
- Performance optimized

Technology Stack

- **React** ^18.3.1
- **TypeScript** ^5.6.3
- **Tailwind CSS** ^3.4.15
- **Vite** ^6.0.1
- **@dnd-kit** (drag-and-drop)
- **date-fns** (date manipulation)
- **clsx** (conditional classes)

Getting Started

Installation

```
``bash
npm install
``
```

Development

```
```bash
npm run dev
```
```

Open [http://localhost:5173] (http://localhost:5173) in your browser.

Build

```
```bash
npm run build
```
```

Preview Production Build

```
```bash
npm run preview
```
```

Project Structure

```
...
kanban-component/
├── src/
│   ├── components/
│   │   ├── KanbanBoard/
│   │   │   ├── KanbanBoard.tsx      # Main board component
│   │   │   ├── KanbanBoard.types.ts # TypeScript interfaces
│   │   │   ├── KanbanColumn.tsx     # Column component
│   │   │   ├── KanbanCard.tsx       # Task card component
│   │   │   └── TaskModal.tsx        # Task edit modal
│   │   └── primitives/
│   │       ├── Button.tsx           # Reusable button
│   │       ├── Modal.tsx            # Reusable modal
│   │       └── Avatar.tsx            # Avatar component
│   ├── utils/
│   │   ├── task.utils.ts            # Task helper functions
│   │   └── column.utils.ts          # Column helper functions
│   ├── styles/
│   │   └── globals.css               # Global styles
│   ├── App.tsx                       # Main app component
│   └── main.tsx                       # Entry point
├── index.html
├── package.json
├── tsconfig.json
├── tailwind.config.js
└── vite.config.ts
...
```

Usage

Basic Implementation

```
```tsx
import { KanbanBoard } from './components/KanbanBoard/KanbanBoard';
import { KanbanColumn, KanbanTask } from
'./components/KanbanBoard/KanbanBoard.types';

const columns: KanbanColumn[] = [
```

```

 {
 id: 'todo',
 title: 'To Do',
 color: 'bg-blue-600',
 taskIds: [],
 },
 // ... more columns
];

 const tasks: Record<string, KanbanTask> = {
 // ... your tasks
 };

 function MyApp() {
 return (
 <KanbanBoard
 columns={columns}
 tasks={tasks}
 onTaskMove={handleTaskMove}
 onTaskCreate={handleTaskCreate}
 onTaskUpdate={handleTaskUpdate}
 onTaskDelete={handleTaskDelete}
 />
);
 }
 ...

```

## ## Accessibility

This Kanban board is built with accessibility as a top priority:

- ✓ Full keyboard navigation
- ✓ Screen reader support (ARIA labels)
- ✓ Focus management
- ✓ Color contrast compliance (WCAG 2.1 AA)
- ✓ Semantic HTML

## ### Keyboard Shortcuts

- **\*\*Tab/Shift+Tab\*\***: Navigate between interactive elements
- **\*\*Enter/Space\*\***: Pick up or activate focused element
- **\*\*Escape\*\***: Close modal or cancel drag
- **\*\*Arrow Keys\*\***: Navigate between cards (during drag)

## ## Performance

- React.memo for component optimization
- Efficient drag-and-drop handling
- Optimized re-renders
- Production bundle size < 200KB (gzipped)

## ## Customization

### ### Adding Custom Columns

Modify the `initialColumns` array in `App.tsx`:

```
```tsx
const columns: KanbanColumn[] = [
  {
    id: 'custom-status',
    title: 'Custom Status',
    color: 'bg-indigo-600',
    taskIds: [],
    maxTasks: 8, // Optional WIP limit
  },
];
```
```

### ### Styling

Customize colors in `tailwind.config.js`:

```
```js
theme: {
  extend: {
    colors: {
      primary: {
        500: '#your-color',
        // ...
      },
    },
  },
},
}
```
```

### ## Browser Support

- Chrome (latest)
- Firefox (latest)
- Safari (latest)
- Edge (latest)

### ## License

MIT

### ## Author

Built with ❤️ following enterprise-grade standards