File: client/src/components/ui/context-menu.tsx

Type: tsx | Size: 7428 bytes

```tsx
import * as React from "react"
import * as ContextMenuPrimitive from "@radix-ui/react-context-menu"
import { Check, ChevronRight, Circle } from "lucide-react"

import { cn } from "@/lib/utils"

const ContextMenu = ContextMenuPrimitive.Root

const ContextMenuTrigger = ContextMenuPrimitive.Trigger

const ContextMenuGroup = ContextMenuPrimitive.Group

const ContextMenuPortal = ContextMenuPrimitive.Portal

const ContextMenuSub = ContextMenuPrimitive.Sub

const ContextMenuRadioGroup = ContextMenuPrimitive.RadioGroup

const ContextMenuSubTrigger = React.forwardRef<
  React.ElementRef<typeof ContextMenuPrimitive.SubTrigger>,
  React.ComponentPropsWithoutRef<typeof ContextMenuPrimitive.SubTrigger> & {
    inset?: boolean
  }
>(({ className, inset, children, ...props }, ref) => (
  <ContextMenuPrimitive.SubTrigger
    ref={ref}
    className={cn(
      "flex cursor-default select-none items-center rounded-sm px-2 py-1.5 text-sm outline-none focus:bg-accen...
      inset && "pl-8",
      className
    )}
    {...props}
  >
    {children}
    <ChevronRight className="ml-auto h-4 w-4" />
  </ContextMenuPrimitive.SubTrigger>
))
ContextMenuSubTrigger.displayName = ContextMenuPrimitive.SubTrigger.displayName

const ContextMenuSubContent = React.forwardRef<
  React.ElementRef<typeof ContextMenuPrimitive.SubContent>,
  React.ComponentPropsWithoutRef<typeof ContextMenuPrimitive.SubContent>
>(({ className, ...props }, ref) => (
  <ContextMenuPrimitive.SubContent
    ref={ref}
    className={cn(
      "z-50 min-w-[8rem] overflow-hidden rounded-md border bg-popover p-1 text-popover-foreground shadow-md da...
      className
    )}
    {...props}
  />
))
ContextMenuSubContent.displayName = ContextMenuPrimitive.SubContent.displayName

const ContextMenuContent = React.forwardRef<
  React.ElementRef<typeof ContextMenuPrimitive.Content>,
  React.ComponentPropsWithoutRef<typeof ContextMenuPrimitive.Content>
>(({ className, ...props }, ref) => (
  <ContextMenuPrimitive.Portal>
    <ContextMenuPrimitive.Content
      ref={ref}
      className={cn(
        "z-50 max-h-[--radix-context-menu-content-available-height] min-w-[8rem] overflow-y-auto overflow-x-hi...
        className
      )}
      {...props}
    />
  </ContextMenuPrimitive.Portal>
))
ContextMenuContent.displayName = ContextMenuPrimitive.Content.displayName
```

```
const ContextMenuItem = React.forwardRef<
  React.ElementRef<typeof ContextMenuPrimitive.Item>,
  React.ComponentPropsWithoutRef<typeof ContextMenuPrimitive.Item> & {
    inset?: boolean
  }
>(({ className, inset, ...props }, ref) => (
  <ContextMenuPrimitive.Item
    ref={ref}
    className={cn(
      "relative flex cursor-default select-none items-center rounded-sm px-2 py-1.5 text-sm outline-none focus...
      inset && "pl-8",
      className
    )}
    {...props}
  />
))
ContextMenuItem.displayName = ContextMenuPrimitive.Item.displayName

const ContextMenuCheckboxItem = React.forwardRef<
  React.ElementRef<typeof ContextMenuPrimitive.CheckboxItem>,
  React.ComponentPropsWithoutRef<typeof ContextMenuPrimitive.CheckboxItem>
>(({ className, children, checked, ...props }, ref) => (
  <ContextMenuPrimitive.CheckboxItem
    ref={ref}
    className={cn(
      "relative flex cursor-default select-none items-center rounded-sm py-1.5 pl-8 pr-2 text-sm outline-none ...
      className
    )}
    checked={checked}
    {...props}
  >
    <span className="absolute left-2 flex h-3.5 w-3.5 items-center justify-center">
      <ContextMenuPrimitive.ItemIndicator>
        <Check className="h-4 w-4" />
      </ContextMenuPrimitive.ItemIndicator>
    </span>
    {children}
  </ContextMenuPrimitive.CheckboxItem>
))
ContextMenuCheckboxItem.displayName =
  ContextMenuPrimitive.CheckboxItem.displayName

const ContextMenuRadioItem = React.forwardRef<
  React.ElementRef<typeof ContextMenuPrimitive.RadioItem>,
  React.ComponentPropsWithoutRef<typeof ContextMenuPrimitive.RadioItem>
>(({ className, children, ...props }, ref) => (
  <ContextMenuPrimitive.RadioItem
    ref={ref}
    className={cn(
      "relative flex cursor-default select-none items-center rounded-sm py-1.5 pl-8 pr-2 text-sm outline-none ...
      className
    )}
    {...props}
  >
    <span className="absolute left-2 flex h-3.5 w-3.5 items-center justify-center">
      <ContextMenuPrimitive.ItemIndicator>
        <Circle className="h-2 w-2 fill-current" />
      </ContextMenuPrimitive.ItemIndicator>
    </span>
    {children}
  </ContextMenuPrimitive.RadioItem>
))
ContextMenuRadioItem.displayName = ContextMenuPrimitive.RadioItem.displayName

const ContextMenuLabel = React.forwardRef<
  React.ElementRef<typeof ContextMenuPrimitive.Label>,
  React.ComponentPropsWithoutRef<typeof ContextMenuPrimitive.Label> & {
    inset?: boolean
  }
>(({ className, inset, ...props }, ref) => (
  <ContextMenuPrimitive.Label
    ref={ref}
    className={cn(
      "px-2 py-1.5 text-sm font-semibold text-foreground",
      inset && "pl-8",
      className
    )}
    {...props}
  />
))
ContextMenuLabel.displayName = ContextMenuPrimitive.Label.displayName

const ContextMenuSeparator = React.forwardRef<
  React.ElementRef<typeof ContextMenuPrimitive.Separator>,
  React.ComponentPropsWithoutRef<typeof ContextMenuPrimitive.Separator>
>(({ className, ...props }, ref) => (
  <ContextMenuPrimitive.Separator
    ref={ref}
    className={cn("-mx-1 my-1 h-px bg-border", className)}
    {...props}
  />
))
ContextMenuSeparator.displayName = ContextMenuPrimitive.Separator.displayName

const ContextMenuShortcut = ({
  className,
  ...props
}: React.HTMLAttributes<HTMLSpanElement>) => {
  return (
    <span
      className={cn(
        "ml-auto text-xs tracking-widest text-muted-foreground",
        className
      )}
      {...props}
    />
  )
}
ContextMenuShortcut.displayName = "ContextMenuShortcut"

export {
  ContextMenu,
  ContextMenuTrigger,
  ContextMenuContent,
  ContextMenuItem,
  ContextMenuCheckboxItem,
  ContextMenuRadioItem,
  ContextMenuLabel,
  ContextMenuSeparator,
  ContextMenuShortcut,
  ContextMenuGroup,
  ContextMenuPortal,
  ContextMenuSub,
  ContextMenuSubContent,
  ContextMenuSubTrigger,
  ContextMenuRadioGroup,
}
```