pokemon-card.md 2025-06-27

```
import { Card, CardContent } from "@/components/ui/card"
import { Button } from "@/components/ui/button"
import { Badge } from "@/components/ui/badge"
import { Plus } from "lucide-react"
import { cn } from "@/lib/utils"
import { Pokemon } from "@/lib/types"
// Props interface for the PokemonCard component
interface PokemonCardProps {
 pokemon: Pokemon
 onAdd: () => void
 onSelect: () => void
}
/**
 * PokemonCard component to display individual Pokémon with their details
* @param pokemon - The Pokémon object to display.
 * @param onAdd - Callback function to handle adding the Pokémon to a team.
 * @param onSelect - Callback function to handle selecting the Pokémon for further
actions.
 * @returns
export default function PokemonCard({ pokemon, onAdd, onSelect }:
PokemonCardProps) {
 // Define colors for different Pokémon types
 const typeColors: Record<string, string> = {
    normal: "bg-stone-400",
    fire: "bg-orange-500",
    water: "bg-blue-500",
    electric: "bg-yellow-400",
    grass: "bg-green-500",
    ice: "bg-cyan-300",
    fighting: "bg-red-700",
    poison: "bg-purple-600",
    ground: "bg-amber-600",
    flying: "bg-indigo-300",
    psychic: "bg-pink-500",
    bug: "bg-lime-500",
    rock: "bg-yellow-700",
    ghost: "bg-purple-800",
    dragon: "bg-violet-700",
    dark: "bg-stone-700",
    steel: "bg-slate-400",
   fairy: "bg-pink-300",
  }
 // If no Pokémon is provided, return null
 if (!pokemon) {
   return null
  }
```

pokemon-card.md 2025-06-27

```
// Extract Pokémon details with fallback values
 const pokemonName = pokemon.name || "Unknown"
 const pokemonTypes = pokemon.types || []
 const pokemonSprite = pokemon.sprite || "/placeholder.svg?height=120&width=120"
 return (
   // Card component to display Pokémon details
   <Card className="overflow-hidden transition-all hover:shadow-md cursor-
pointer" onClick={onSelect}>
      <div className="relative bg-slate-100 dark:bg-slate-800 pt-2">
        <div className="absolute top-2 right-2 z-10">
          {/* Button to add Pokémon to the team */}
          <Button
            variant="ghost"
            size="icon"
            className="h-7 w-7 rounded-full bg-white/80 hover:bg-white shadow-sm"
            onClick={(e) => {
              e.stopPropagation()
              onAdd()
            }}
            <Plus className="h-4 w-4" />
            <span className="sr-only">Add to team</span>
          </Button>
        </div>
        {/* Pokémon sprite image */}
        <div className="flex justify-center">
          <img
            src={pokemonSprite | | "/placeholder.svg"}
            alt={pokemonName}
            className="h-[120px] w-[120px] object-contain"
            onError={(e) => {
              const target = e.target as HTMLImageElement
              target.src = "/placeholder.svg?height=120&width=120"
            }}
          />
        </div>
      </div>
      {/* Card content displaying Pokémon name and types */}
      <CardContent className="p-3">
        <h3 className="font-medium text-sm truncate text-center">{pokemonName}
</h3>
        <div className="flex justify-center gap-1 mt-1">
          {pokemonTypes.map((type: string, index: number) => (
            <Badge
              key={`${type}-${index}`}
              variant="secondary"
              className={cn("text-white text-xs", typeColors[type?.toLowerCase()]
|| "bg-gray-500")}
            >
              {type}
            </Badge>
          ))}
        </div>
```

pokemon-card.md 2025-06-27

```
</CardContent>
</Card>
)
}
```