

Build this as my initial prototype

Copy-paste this component to /components/ui folder:

```
``tsx
```

```
footer-section.tsx
```

```
"use client"
```

```
import * as React from "react"
```

```
import { Button } from "@components/ui/button"
```

```
import { Input } from "@components/ui/input"
```

```
import { Label } from "@components/ui/label"
```

```
import { Switch } from "@components/ui/switch"
```

```
import { Textarea } from "@components/ui/textarea"
```

```
import {
```

```
  Tooltip,
```

```
  TooltipContent,
```

```
  TooltipProvider,
```

```
  TooltipTrigger,
```

```
} from "@components/ui/tooltip"
```

```
import { Facebook, Instagram, Linkedin, Moon, Send, Sun, Twitter } from "lucide-react"
```

```
function Footerdemo() {
```

```
  const [isDarkMode, setIsDarkMode] = React.useState(true)
```

```
  const [isChatOpen, setIsChatOpen] = React.useState(false)
```

```
  React.useEffect(() => {
```

```
    if (isDarkMode) {
```

```
      document.documentElement.classList.add("dark")
```

```
    } else {
```

```
      document.documentElement.classList.remove("dark")
```

```
    }
```

```
  }, [isDarkMode])
```

```
  return (
```

```
    <footer className="relative border-t bg-background text-foreground transition-colors duration-300">
```

```
      <div className="container mx-auto px-4 py-12 md:px-6 lg:px-8">
```

```
        <div className="grid gap-12 md:grid-cols-2 lg:grid-cols-4">
```

```
          <div className="relative">
```

```
            <h2 className="mb-4 text-3xl font-bold tracking-tight">Stay Connected</h2>
```

```
            <p className="mb-6 text-muted-foreground">
```

```
              Join our newsletter for the latest updates and exclusive offers.
```

```
            </p>
```

```
            <form className="relative">
```

```

    <Input
      type="email"
      placeholder="Enter your email"
      className="pr-12 backdrop-blur-sm"
    />
    <Button
      type="submit"
      size="icon"
      className="absolute right-1 top-1 h-8 w-8 rounded-full bg-primary
text-primary-foreground transition-transform hover:scale-105"
    >
      <Send className="h-4 w-4" />
      <span className="sr-only">Subscribe</span>
    </Button>
  </form>
  <div className="absolute -right-4 top-0 h-24 w-24 rounded-full bg-primary/10 blur-2xl"
/>
</div>
<div>
  <h3 className="mb-4 text-lg font-semibold">Quick Links</h3>
  <nav className="space-y-2 text-sm">
    <a href="#" className="block transition-colors hover:text-primary">
      Home
    </a>
    <a href="#" className="block transition-colors hover:text-primary">
      About Us
    </a>
    <a href="#" className="block transition-colors hover:text-primary">
      Services
    </a>
    <a href="#" className="block transition-colors hover:text-primary">
      Products
    </a>
    <a href="#" className="block transition-colors hover:text-primary">
      Contact
    </a>
  </nav>
</div>
<div>
  <h3 className="mb-4 text-lg font-semibold">Contact Us</h3>
  <address className="space-y-2 text-sm not-italic">
    <p>123 Innovation Street</p>
    <p>Tech City, TC 12345</p>
    <p>Phone: (123) 456-7890</p>
  </address>
</div>

```

```
<p>Email: hello@example.com</p>
</address>
</div>
<div className="relative">
  <h3 className="mb-4 text-lg font-semibold">Follow Us</h3>
  <div className="mb-6 flex space-x-4">
    <TooltipProvider>
      <Tooltip>
        <TooltipTrigger asChild>
          <Button variant="outline" size="icon" className="rounded-full">
            <Facebook className="h-4 w-4" />
            <span className="sr-only">Facebook</span>
          </Button>
        </TooltipTrigger>
        <TooltipContent>
          <p>Follow us on Facebook</p>
        </TooltipContent>
      </Tooltip>
    </TooltipProvider>
    <TooltipProvider>
      <Tooltip>
        <TooltipTrigger asChild>
          <Button variant="outline" size="icon" className="rounded-full">
            <Twitter className="h-4 w-4" />
            <span className="sr-only">Twitter</span>
          </Button>
        </TooltipTrigger>
        <TooltipContent>
          <p>Follow us on Twitter</p>
        </TooltipContent>
      </Tooltip>
    </TooltipProvider>
    <TooltipProvider>
      <Tooltip>
        <TooltipTrigger asChild>
          <Button variant="outline" size="icon" className="rounded-full">
            <Instagram className="h-4 w-4" />
            <span className="sr-only">Instagram</span>
          </Button>
        </TooltipTrigger>
        <TooltipContent>
          <p>Follow us on Instagram</p>
        </TooltipContent>
      </Tooltip>
    </TooltipProvider>
  </div>
</div>
```

```

</TooltipProvider>
<TooltipProvider>
  <Tooltip>
    <TooltipTrigger asChild>
      <Button variant="outline" size="icon" className="rounded-full">
        <Linkedin className="h-4 w-4" />
        <span className="sr-only">LinkedIn</span>
      </Button>
    </TooltipTrigger>
    <TooltipContent>
      <p>Connect with us on LinkedIn</p>
    </TooltipContent>
  </Tooltip>
</TooltipProvider>
</div>
<div className="flex items-center space-x-2">
  <Sun className="h-4 w-4" />
  <Switch
    id="dark-mode"
    checked={isDarkMode}
    onCheckedChange={setIsDarkMode}
  />
  <Moon className="h-4 w-4" />
  <Label htmlFor="dark-mode" className="sr-only">
    Toggle dark mode
  </Label>
</div>
</div>
</div>
<div className="mt-12 flex flex-col items-center justify-between gap-4 border-t pt-8
text-center md:flex-row">
  <p className="text-sm text-muted-foreground">
    © 2024 Your Company. All rights reserved.
  </p>
  <nav className="flex gap-4 text-sm">
    <a href="#" className="transition-colors hover:text-primary">
      Privacy Policy
    </a>
    <a href="#" className="transition-colors hover:text-primary">
      Terms of Service
    </a>
    <a href="#" className="transition-colors hover:text-primary">
      Cookie Settings
    </a>
  </nav>
</div>

```

```

        </nav>
      </div>
    </div>
  </footer>
)
}

```

```
export { Footerdemo }
```

demo.tsx

```
import { Footerdemo } from "@components/ui/footer-section";
```

```
function Footer() {
  return (
    <div className="block">
      <Footerdemo />
    </div>
  );
}

```

```
export { Footer };
```

```
...
```

Copy-paste these files for dependencies:

```

```tsx
shadcn/button
import * as React from "react"
import { Slot } from "@radix-ui/react-slot"
import { cva, type VariantProps } from "class-variance-authority"

import { cn } from "@lib/utils"

```

```

const buttonVariants = cva(
 "inline-flex items-center justify-center whitespace-nowrap rounded-md text-sm font-medium
 ring-offset-background transition-colors focus-visible:outline-none focus-visible:ring-2
 focus-visible:ring-ring focus-visible:ring-offset-2 disabled:pointer-events-none
 disabled:opacity-50",
 {
 variants: {
 variant: {
 default: "bg-primary text-primary-foreground hover:bg-primary/90",
 destructive:
 "bg-destructive text-destructive-foreground hover:bg-destructive/90",

```

```

 outline:
 "border border-input bg-background hover:bg-accent hover:text-accent-foreground",
 secondary:
 "bg-secondary text-secondary-foreground hover:bg-secondary/80",
 ghost: "hover:bg-accent hover:text-accent-foreground",
 link: "text-primary underline-offset-4 hover:underline",
 },
 size: {
 default: "h-10 px-4 py-2",
 sm: "h-9 rounded-md px-3",
 lg: "h-11 rounded-md px-8",
 icon: "h-10 w-10",
 },
},
defaultVariants: {
 variant: "default",
 size: "default",
},
},
)

```

```

export interface ButtonProps
 extends React.ButtonHTMLAttributes<HTMLButtonElement>,
 VariantProps<typeof buttonVariants> {
 asChild?: boolean
}

```

```

const Button = React.forwardRef<HTMLButtonElement, ButtonProps>(
 ({ className, variant, size, asChild = false, ...props }, ref) => {
 const Comp = asChild ? Slot : "button"
 return (
 <Comp
 className={cn(buttonVariants({ variant, size, className }))}
 ref={ref}
 {...props}
 />
)
 },
)
Button.displayName = "Button"

```

```

export { Button, buttonVariants }

```

```

...

```

```

```tsx
shadcn/input
import * as React from "react"

import { cn } from "@lib/utls"

export interface InputProps
  extends React.HTMLAttributes<HTMLInputElement> {}

const Input = React.forwardRef<HTMLInputElement, InputProps>(
  ({ className, type, ...props }, ref) => {
    return (
      <input
        type={type}
        className={cn(
          "flex h-10 w-full rounded-md border border-input bg-background px-3 py-2 text-sm
ring-offset-background file:border-0 file:bg-transparent file:text-sm file:font-medium
file:text-foreground placeholder:text-muted-foreground focus-visible:outline-none
focus-visible:ring-2 focus-visible:ring-ring focus-visible:ring-offset-2 disabled:cursor-not-allowed
disabled:opacity-50",
          className
        )}
        ref={ref}
        {...props}
      />
    )
  }
)
Input.displayName = "Input"

export { Input }

...
```tsx
shadcn/label
"use client"

import * as React from "react"
import * as LabelPrimitive from "@radix-ui/react-label"
import { cva, type VariantProps } from "class-variance-authority"

import { cn } from "@lib/utls"

const labelVariants = cva(

```

```
"text-sm font-medium leading-none peer-disabled:cursor-not-allowed
peer-disabled:opacity-70",
)
```

```
const Label = React.forwardRef<
 React.ElementRef<typeof LabelPrimitive.Root>,
 React.ComponentPropsWithoutRef<typeof LabelPrimitive.Root> &
 VariantProps<typeof labelVariants>
>(({ className, ...props }, ref) => (
 <LabelPrimitive.Root
 ref={ref}
 className={cn(labelVariants(), className)}
 {...props}
 />
))
Label.displayName = LabelPrimitive.Root.displayName
```

```
export { Label }
```

```
...
```

```
```tsx
```

```
shadcn/switch
```

```
"use client"
```

```
import * as React from "react"
```

```
import * as SwitchPrimitives from "@radix-ui/react-switch"
```

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

```
const Switch = React.forwardRef<
  React.ElementRef<typeof SwitchPrimitives.Root>,
  React.ComponentPropsWithoutRef<typeof SwitchPrimitives.Root>
>(({ className, ...props }, ref) => (
  <SwitchPrimitives.Root
    className={cn(
      "peer inline-flex h-6 w-11 shrink-0 cursor-pointer items-center rounded-full border-2
border-transparent transition-colors focus-visible:outline-none focus-visible:ring-2
focus-visible:ring-ring focus-visible:ring-offset-2 focus-visible:ring-offset-background
disabled:cursor-not-allowed disabled:opacity-50 data-[state=checked]:bg-primary
data-[state=unchecked]:bg-input",
      className,
    )}
    {...props}
    ref={ref}
```



```

>
  <SwitchPrimitives.Thumb
    className={cn(
      "pointer-events-none block h-5 w-5 rounded-full bg-background shadow-lg ring-0
transition-transform data-[state=checked]:translate-x-5 data-[state=unchecked]:translate-x-0",
    )}
  />
</SwitchPrimitives.Root>
))
Switch.displayName = SwitchPrimitives.Root.displayName

export { Switch }

...
```tsx
shadcn/textarea
import * as React from "react"

import { cn } from "@lib/Utils"

export interface TextareaProps
 extends React.TextareaHTMLAttributes<HTMLTextAreaElement> {}

const Textarea = React.forwardRef<HTMLTextAreaElement, TextareaProps>(
 ({ className, ...props }, ref) => {
 return (
 <textarea
 className={cn(
 "flex min-h-[80px] w-full rounded-md border border-input bg-background px-3 py-2
text-sm ring-offset-background placeholder:text-muted-foreground focus-visible:outline-none
focus-visible:ring-2 focus-visible:ring-ring focus-visible:ring-offset-2 disabled:cursor-not-allowed
disabled:opacity-50",
 className
)}
 ref={ref}
 {...props}
 />
)
 }
)
Textarea.displayName = "Textarea"

export { Textarea }

```

```

...
```tsx
shadcn/tooltip
"use client"

import * as React from "react"
import * as TooltipPrimitive from "@radix-ui/react-tooltip"

import { cn } from "@lib/Utils"

const TooltipProvider = TooltipPrimitive.Provider

const Tooltip = TooltipPrimitive.Root

const TooltipTrigger = TooltipPrimitive.Trigger

const TooltipContent = React.forwardRef<
  React.ElementRef<typeof TooltipPrimitive.Content>,
  React.ComponentPropsWithoutRef<typeof TooltipPrimitive.Content>
>(({ className, sideOffset = 4, ...props }, ref) => (
  <TooltipPrimitive.Content
    ref={ref}
    sideOffset={sideOffset}
    className={cn(
      "z-50 overflow-hidden rounded-md border bg-popover px-3 py-1.5 text-sm
text-popover-foreground shadow-md animate-in fade-in-0 zoom-in-95
data-[state=closed]:animate-out data-[state=closed]:fade-out-0 data-[state=closed]:zoom-out-95
data-[side=bottom]:slide-in-from-top-2 data-[side=left]:slide-in-from-right-2
data-[side=right]:slide-in-from-left-2 data-[side=top]:slide-in-from-bottom-2",
      className,
    )}
    {...props}
  />
))
TooltipContent.displayName = TooltipPrimitive.Content.displayName

export { Tooltip, TooltipTrigger, TooltipContent, TooltipProvider }
...

```

Install these NPM dependencies:

```

```bash
lucide-react, @radix-ui/react-slot, class-variance-authority, @radix-ui/react-label,
@radix-ui/react-switch, @radix-ui/react-tooltip

```

