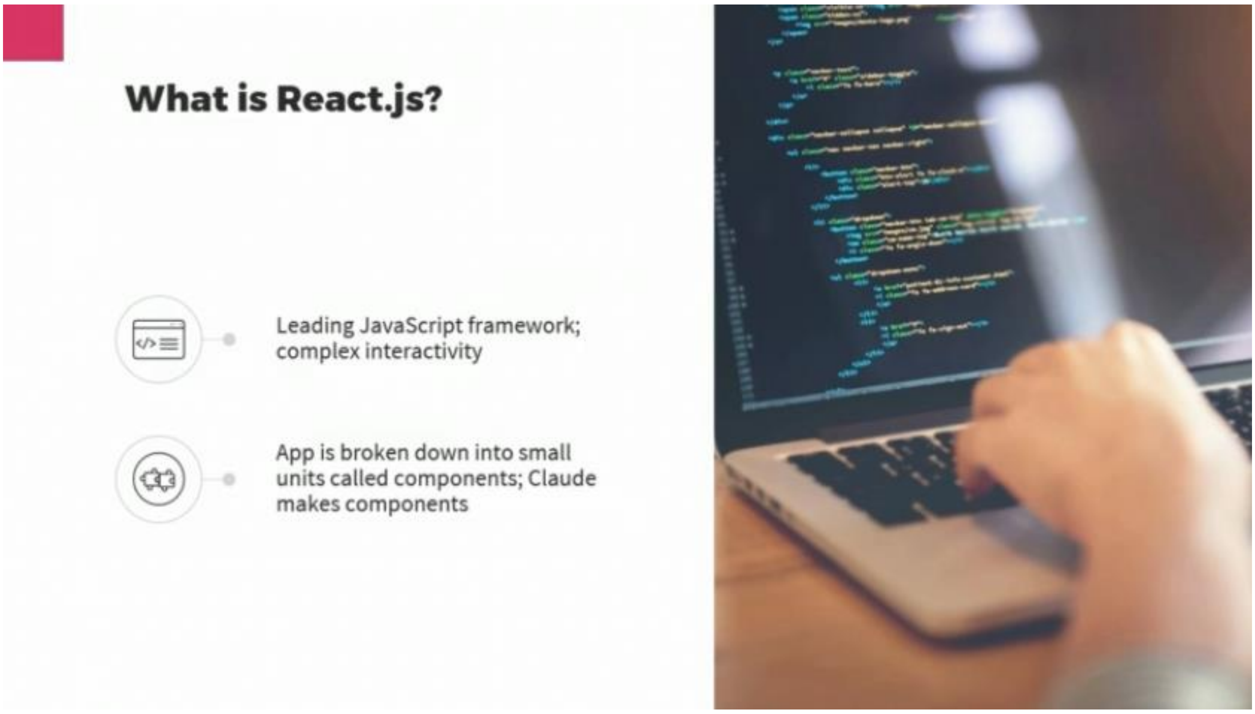
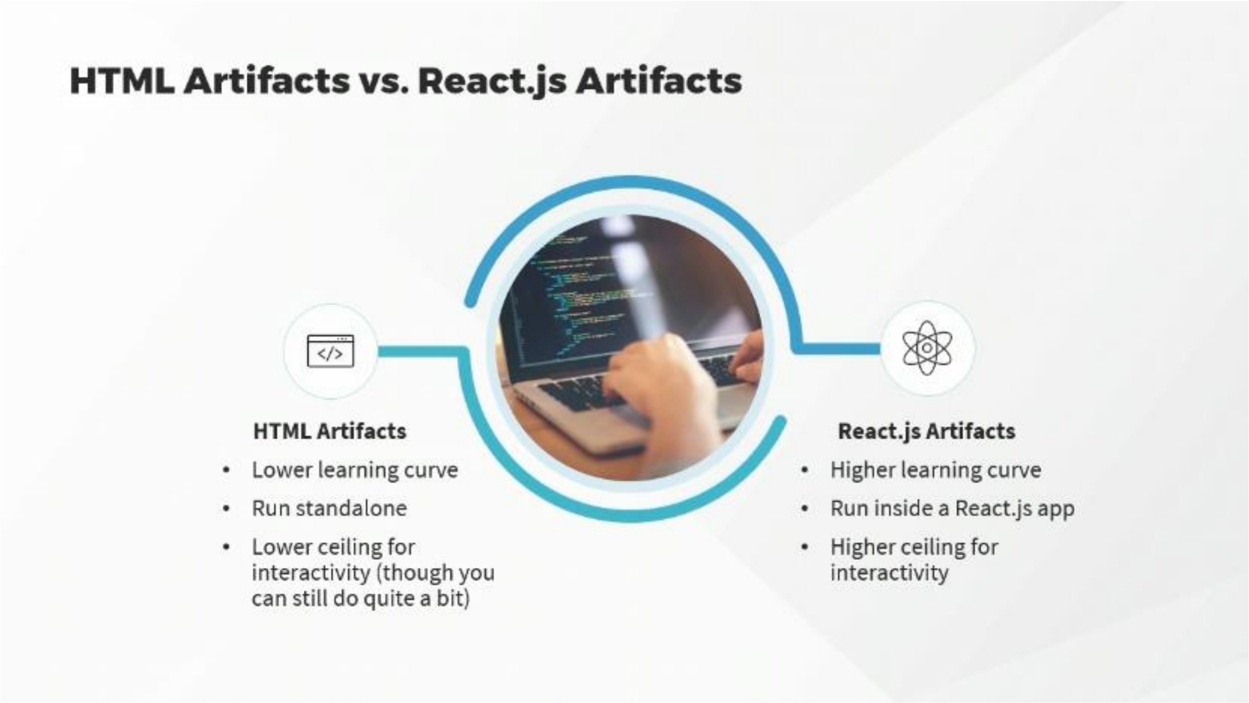
REACT.JS



What is React.js?

- Leading JavaScript framework; complex interactivity
- App is broken down into small units called components; Claude makes components



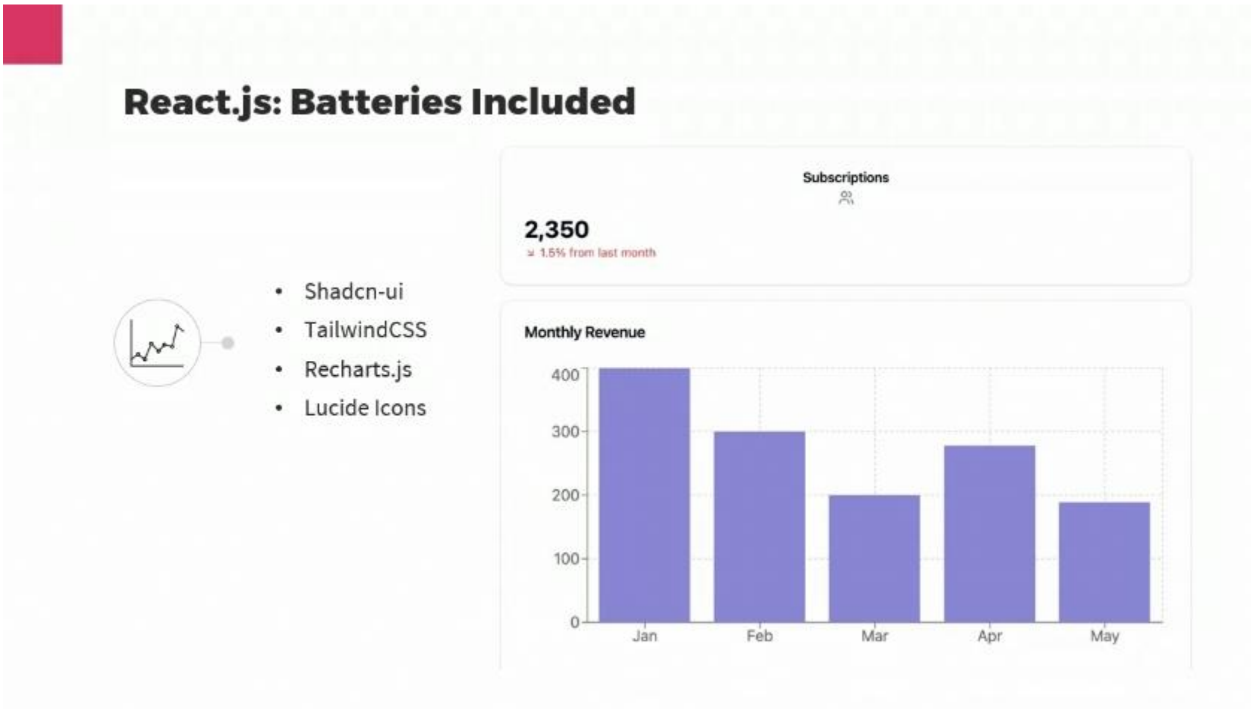
HTML Artifacts vs. React.js Artifacts

HTML Artifacts

- Lower learning curve
- Run standalone
- Lower ceiling for interactivity (though you can still do quite a bit)

React.js Artifacts

- Higher learning curve
- Run inside a React.js app
- Higher ceiling for interactivity

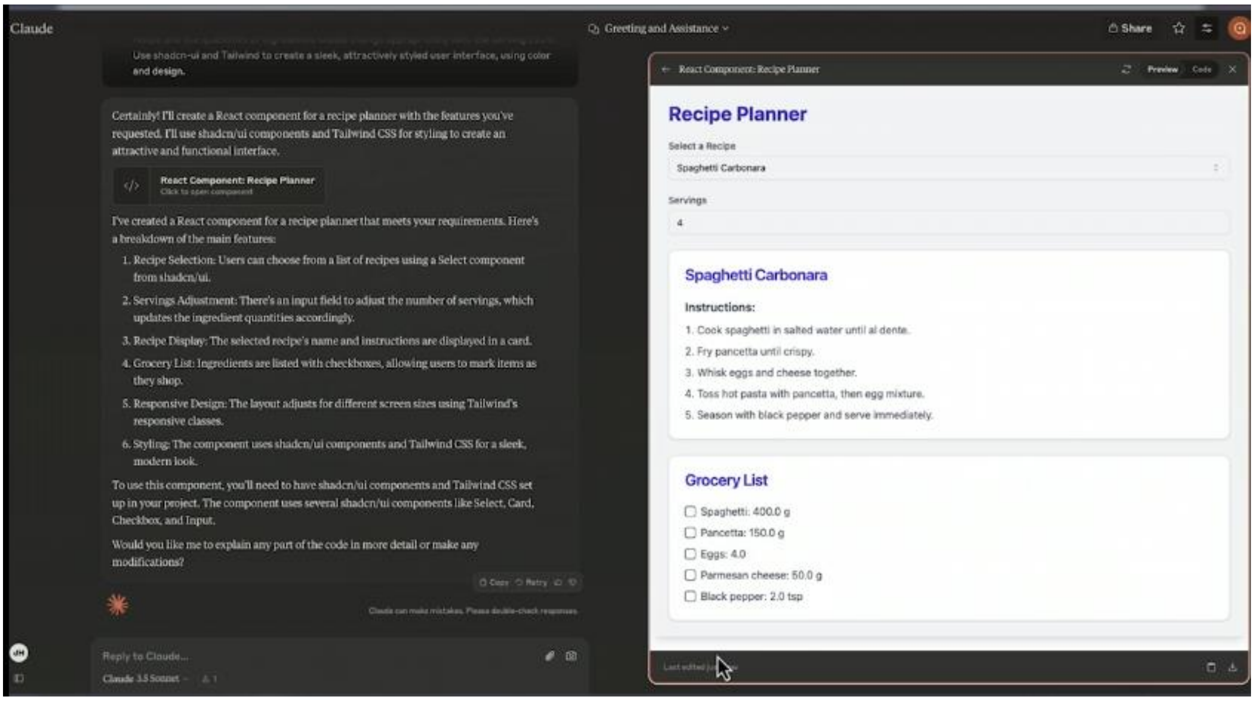


React.js: Batteries Included

- Shadcn-ui
- TailwindCSS
- Recharts.js
- Lucide Icons

Prompt Example:

Generate a React Artifact for a recipe planner. The planner should allow the user to select a recipe, render that recipe, and display a grocery list with checkboxes for all of the recipe's ingredients. There should also be a number input that can set the desired servings of the recipe, and the quantities of ingredients should change appropriately with the serving count. Use shadcn-ui and Tailwind to create a sleek, attractively styled user interface, using color, and design.



Use shadcn-ui and Tailwind to create a sleek, attractively styled user interface, using color and design.

Certainly! I'll create a React component for a recipe planner with the features you've requested. I'll use shadcn/ui components and Tailwind CSS for styling to create an attractive and functional interface.

React Component: Recipe Planner
Click to open component

I've created a React component for a recipe planner that meets your requirements. Here's a breakdown of the main features:

1. Recipe Selection: Users can choose from a list of recipes using a Select component from shadcn/ui.
2. Servings Adjustment: There's an input field to adjust the number of servings, which updates the ingredient quantities accordingly.
3. Recipe Display: The selected recipe's name and instructions are displayed in a card.
4. Grocery List: Ingredients are listed with checkboxes, allowing users to mark items as they shop.
5. Responsive Design: The layout adjusts for different screen sizes using Tailwind's responsive classes.
6. Styling: The component uses shadcn/ui components and Tailwind CSS for a sleek, modern look.

To use this component, you'll need to have shadcn/ui components and Tailwind CSS set up in your project. The component uses several shadcn/ui components like Select, Card, Checkbox, and Input.

Would you like me to explain any part of the code in more detail or make any modifications?

Claude can make mistakes. Please double-check responses.

Recipe Planner

Select a Recipe
Spaghetti Carbonara

Servings
4

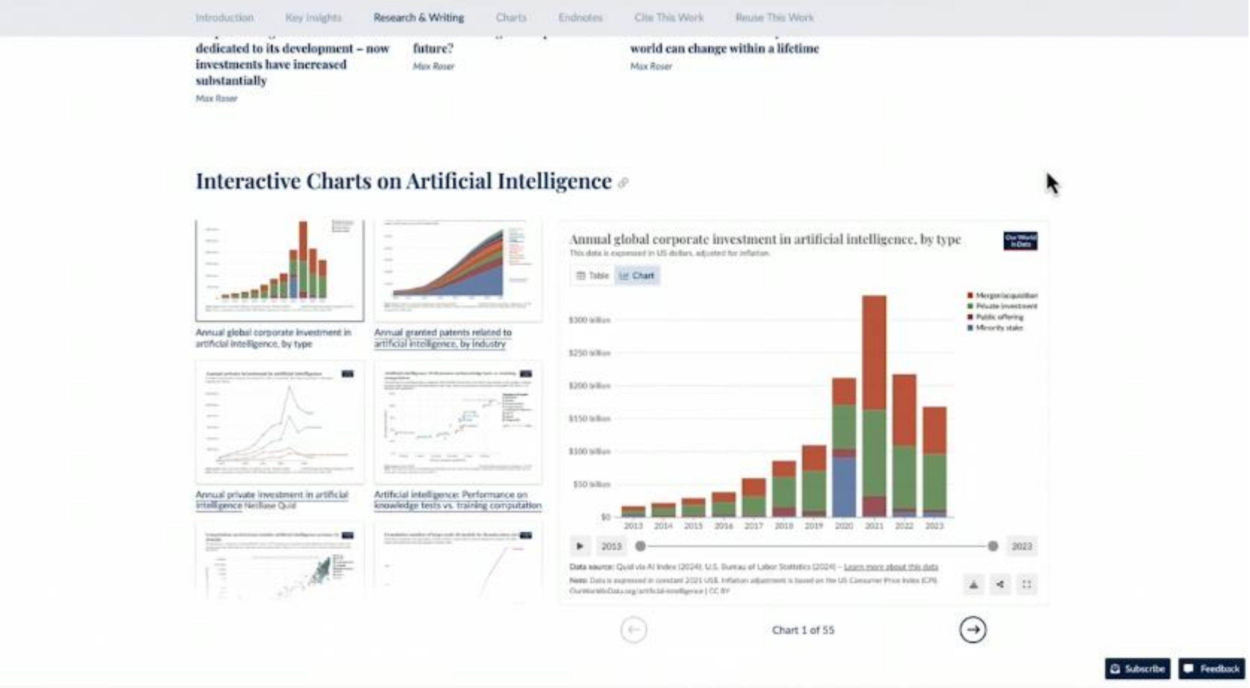
Spaghetti Carbonara

Instructions:

1. Cook spaghetti in salted water until al dente.
2. Fry pancetta until crispy.
3. Whisk eggs and cheese together.
4. Toss hot pasta with pancetta, then egg mixture.
5. Season with black pepper and serve immediately.

Grocery List

☐ Spaghetti: 400.0 g
☐ Pancetta: 150.0 g
☐ Eggs: 4.0
☐ Parmesan cheese: 50.0 g
☐ Black pepper: 2.0 tsp



dedicated to its development – now investments have increased substantially

Max Roser

future?

Max Roser

world can change within a lifetime

Max Roser

Interactive Charts on Artificial Intelligence

Annual global corporate investment in artificial intelligence, by type

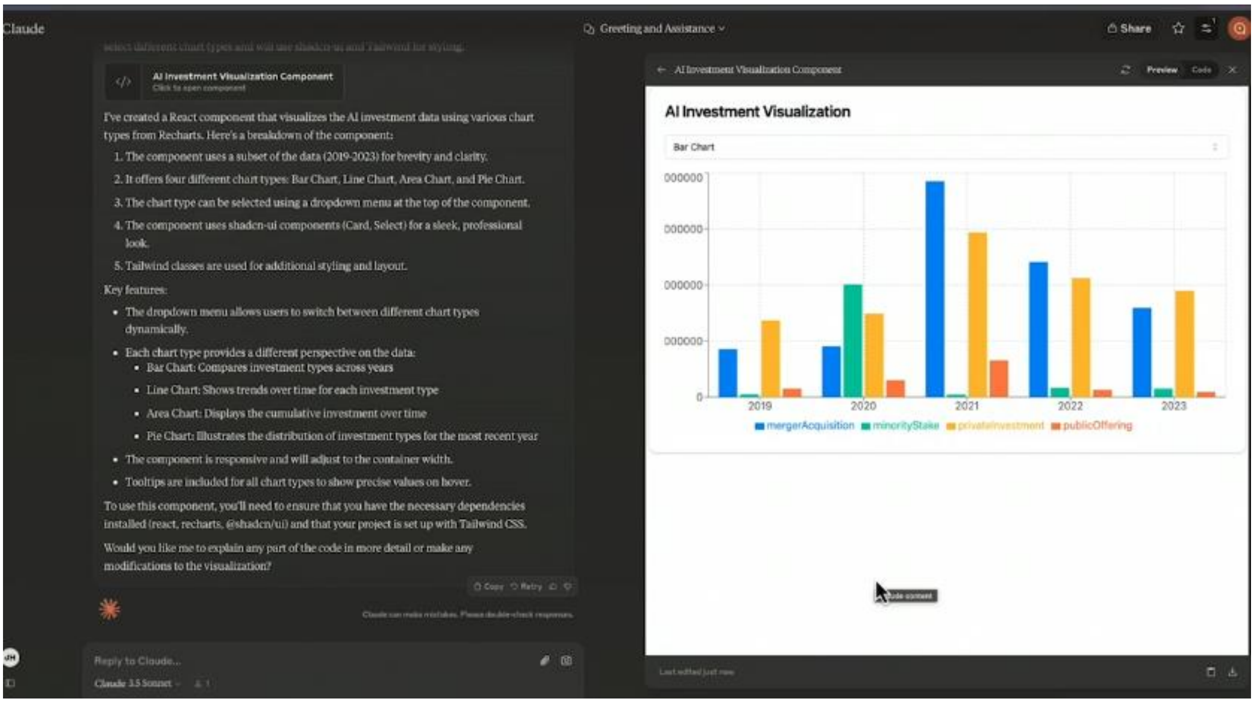
Annual granted patents related to artificial intelligence, by industry

Annual private investment in artificial intelligence NetBase Quid

Artificial intelligence: Performance on knowledge tests vs. training computation

Annual global corporate investment in artificial intelligence, by type

This data is expressed in US dollars, adjusted for inflation.



select different chart types and will use shadcn-ui and Tailwind for styling.

AI Investment Visualization Component
Click to open component

I've created a React component that visualizes the AI investment data using various chart types from Recharts. Here's a breakdown of the component:

1. The component uses a subset of the data (2019-2023) for brevity and clarity.
2. It offers four different chart types: Bar Chart, Line Chart, Area Chart, and Pie Chart.
3. The chart type can be selected using a dropdown menu at the top of the component.
4. The component uses shadcn-ui components (Card, Select) for a sleek, professional look.
5. Tailwind classes are used for additional styling and layout.

Key features:

- The dropdown menu allows users to switch between different chart types dynamically.
- Each chart type provides a different perspective on the data:
  - Bar Chart: Compares investment types across years
  - Line Chart: Shows trends over time for each investment type
  - Area Chart: Displays the cumulative investment over time
  - Pie Chart: Illustrates the distribution of investment types for the most recent year
- The component is responsive and will adjust to the container width.
- Tooltips are included for all chart types to show precise values on hover.

To use this component, you'll need to ensure that you have the necessary dependencies installed (react, recharts, @shadcn/ui) and that your project is set up with Tailwind CSS.

Would you like me to explain any part of the code in more detail or make any modifications to the visualization?

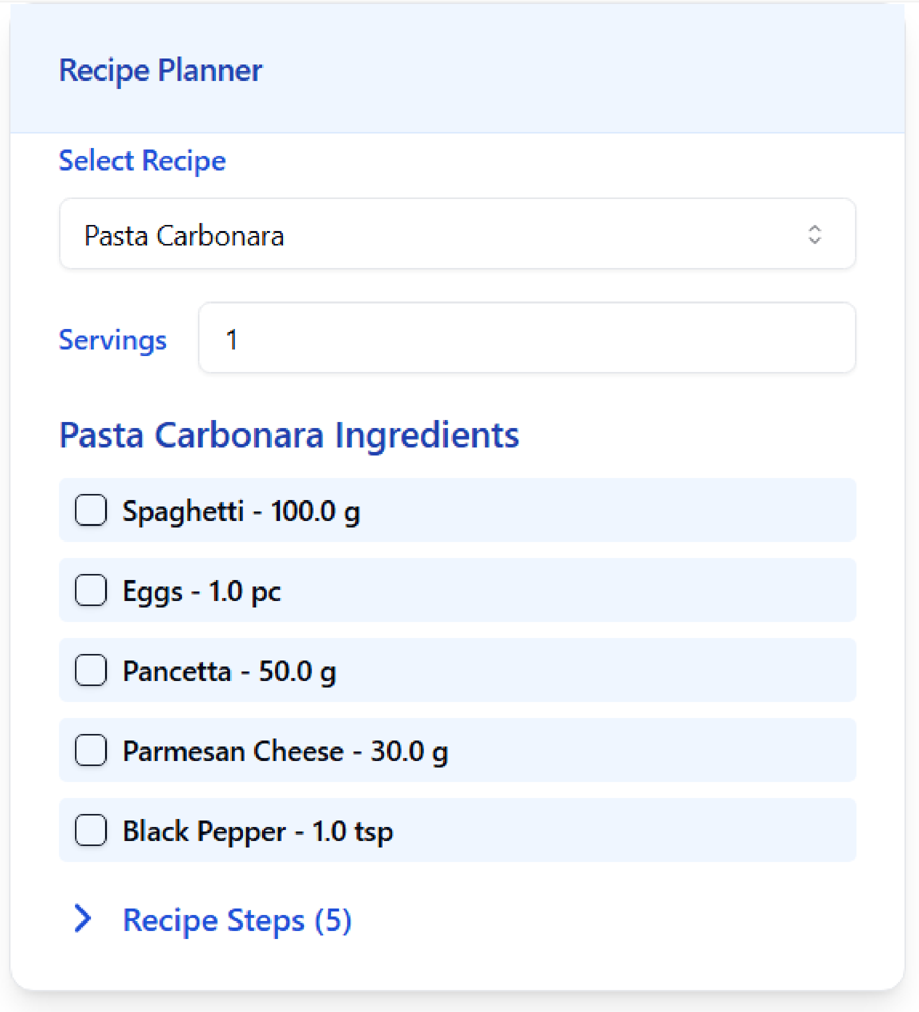
Claude can make mistakes. Please double-check responses.

AI Investment Visualization

Prompt

Generate a React Artifact for a recipe planner. The planner should allow the user to select a recipe, render that recipe, and display a grocery list with checkboxes for all of the recipe's ingredients. There should also be a number input that can set the desired servings of the recipe, and the quantities of ingredients should change appropriately with the serving count. Use shadcn-ui and Tailwind to create a sleek, attractively styled user interface, using color, and design.

The user should be able to select among different recipes.



Recipe Planner

Select Recipe

Pasta Carbonara

Servings 1

Pasta Carbonara Ingredients

☐ Spaghetti - 100.0 g

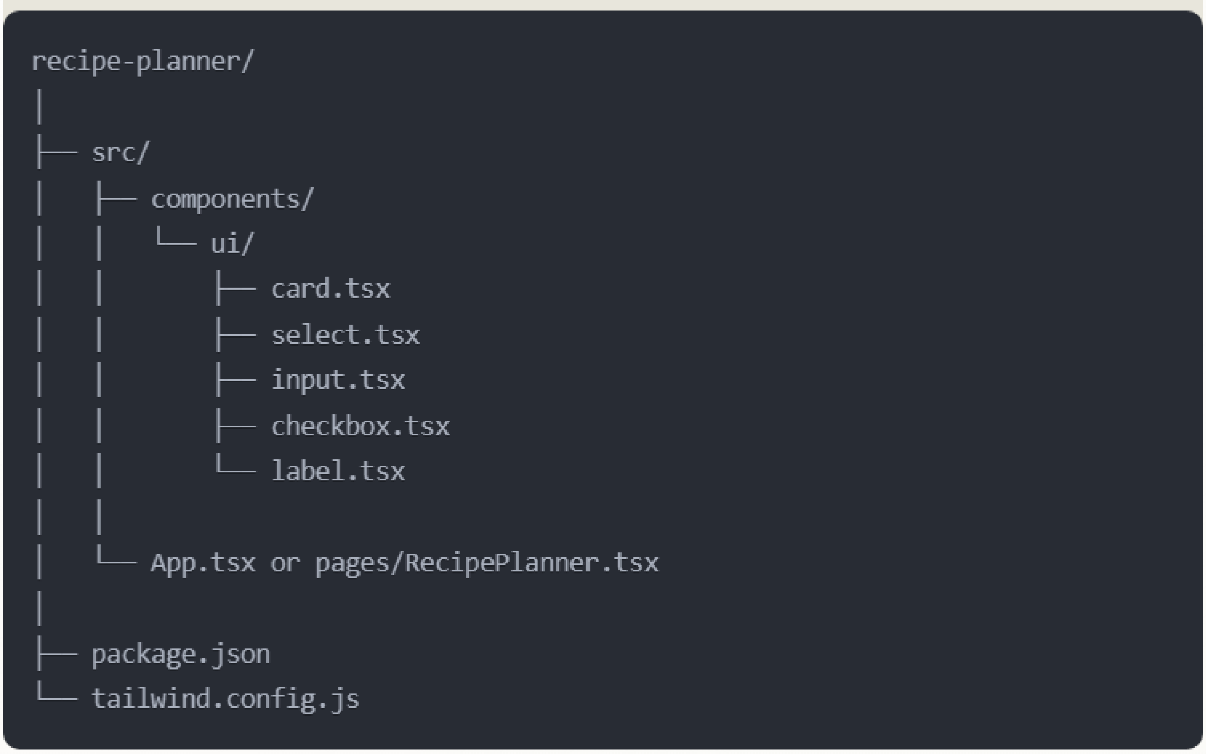
☐ Eggs - 1.0 pc

☐ Pancetta - 50.0 g

☐ Parmesan Cheese - 30.0 g

☐ Black Pepper - 1.0 tsp

Can you give me the folder and file structure that I should create to implement the code you just gave me?



```
recipe-planner/
│
├── src/
│   ├── components/
│   │   └── ui/
│   │       ├── card.tsx
│   │       ├── select.tsx
│   │       ├── input.tsx
│   │       ├── checkbox.tsx
│   │       └── label.tsx
│   │
│   └── App.tsx or pages/RecipePlanner.tsx
│
├── package.json
└── tailwind.config.js
```

Can you give me the code for each file, and tell me how Can I create a react app with Vite that shows this component?

I'll provide a comprehensive setup guide with code for each file.

1. **Create Vite React Project:**

```
npm create vite@latest recipe-planner -- --template react-ts

cd recipe-planner

npm install

npm install @radix-ui/react-select @radix-ui/react-checkbox @radix-ui/react-label lucide-react

npx shadcn-ui@latest init
```

2. **tailwind.config.js:**

```
module.exports = {

  darkMode: ["class"],

  content: [

    "./pages/**/*.{ts,tsx}",

    "./components/**/*.{ts,tsx}",

    "./app/**/*.{ts,tsx}",

    "./src/**/*.{ts,tsx}",
  ],
  prefix: "",
  theme: {
    container: {
      center: true,
      padding: "2rem",
      screens: {
        "2xl": "1400px",
      },
    },
    extend: {
      colors: {
        blue: {
          50: "#eff6ff",
          100: "#dbeafe",
          800: "#1e40af"
        }
      }
    },
  },
  plugins: [require("tailwindcss-animate")],
}
```

3. **package.json (in addition to Vite default, add):**

```
{
  "dependencies": {
    "@radix-ui/react-checkbox": "^1.0.4",
    "@radix-ui/react-label": "^2.0.2",
    "@radix-ui/react-select": "^2.0.0",
    "class-variance-authority": "^0.7.0",
    "clsx": "^2.0.0",
    "lucide-react": "^0.321.0",
    "tailwind-merge": "^2.2.1"
  }
}
```

4. **UI Components (src/components/ui/): These will be generated by shadcn-ui CLI. I'll show a template for card.tsx:**

```
// src/components/ui/card.tsx
import * as React from "react"
import { cn } from "@/lib/utils"

const Card = React.forwardRef
  HTMLDivElement,
  React.HTMLAttributes<HTMLDivElement>
>(({ className, ...props }, ref) => (
  <div
    ref={ref}
    className={cn(
      "rounded-xl border bg-card text-card-foreground shadow",
      className
    )}
    {...props}
  />
))
Card.displayName = "Card"

const CardHeader = React.forwardRef
```

```
  HTMLDivElement,
  React.HTMLAttributes<HTMLDivElement>
>(({ className, ...props }, ref) => (
  <div
    ref={ref}
    className={cn("flex flex-col space-y-1.5 p-6", className)}
    {...props}
  />
))
CardHeader.displayName = "CardHeader"

const CardTitle = React.forwardRef
  HTMLParagraphElement,
  React.HTMLAttributes<HTMLHeadingElement>
>(({ className, ...props }, ref) => (
  <h3
    ref={ref}
    className={cn(
      "text-2xl font-semibold leading-none tracking-tight",
      className
    )}
    {...props}
  />
))
CardTitle.displayName = "CardTitle"

const CardContent = React.forwardRef
  HTMLDivElement,
  React.HTMLAttributes<HTMLDivElement>
```

```
>(({ className, ...props }, ref) => (
  <div ref={ref} className={cn("p-6 pt-0", className)} {...props} />
))
CardContent.displayName = "CardContent"

export { Card, CardHeader, CardTitle, CardContent }
```

5. **src/App.tsx:**

```
import React from 'react'
import RecipePlanner from './RecipePlanner'

function App() {
  return (
    <div className="container mx-auto p-4">
      <RecipePlanner />
    </div>
  )
}

export default App
```

6. **src/RecipePlanner.tsx: Use the component from the previous artifact.**
7. **Install remaining dependencies:**

```
npm install tailwindcss postcss autoprefixer
npx tailwindcss init -p
npm run dev
```

Key shadcn-ui CLI Prompts:

- Typescript: Yes

- Style: Default
- Base color: Slate
- Global CSS: Yes
- Components: Select, Checkbox, Input, Label

This setup provides a complete, production-ready React application with the Recipe Planner component, Tailwind CSS, and shadcn-ui components.

Would you like me to elaborate on any specific part of the setup?