Sunburst Chart Implementation Documentation

Project Overview

This project implements an interactive Sunburst Chart visualization using Chart.js, designed to display hierarchical data in a circular format. The implementation closely mirrors Plotly's sunburst chart features while providing a modern, interactive user experience.

Key Features

1. Interactive Navigation

- Breadcrumb navigation showing the current path in the hierarchy
- Click-to-drill-down functionality for exploring deeper levels
- Smooth transitions between hierarchy levels
- Persistent breadcrumb display for better user orientation

2. Data Visualization

- Multi-level hierarchical data representation
- Automatic color assignment for different hierarchy levels
- Percentage display for each segment
- Segment labels with proper positioning and visibility
- Dynamic sizing based on data values

3. Technical Implementation

Core Components

1. **SunburstChart Component**
    - Main visualization component
    - Handles chart rendering and updates
    - Manages user interactions
    - Implements responsive design
2. **useSunburstNavigation Hook**
    - Manages navigation state
    - Handles path tracking
    - Provides navigation utilities
    - Maintains breadcrumb information
3. **Chart Utilities**
    - Data transformation functions
    - Color management
    - Label positioning
    - Percentage calculations

4. Data Structure

```
{
  name: string,
  value: number,
  children: [
    {
      name: string,
      value: number,
      children: [...]
    }
  ]
}
```

5. Key Features Implementation Details

Breadcrumb Navigation

- Always visible for better user orientation
- Shows complete path from root to current level
- Interactive elements for quick navigation
- Clear visual hierarchy

Segment Information

- Dynamic label positioning
- Percentage calculations relative to parent
- Automatic text wrapping for long labels
- Color-coded segments based on hierarchy level

Color Management

- Automatic color assignment for different levels
- Consistent color scheme throughout the visualization
- High contrast for better readability
- Color persistence across navigation

Technical Decisions

1. Chart.js Selection

- Chosen for its flexibility and performance
- Extensive customization options
- Good community support
- Built-in animation support

2. Component Architecture

- Modular design for better maintainability
- Separation of concerns between visualization and navigation
- Reusable utilities for common operations
- Clean and maintainable code structure

3. Performance Considerations

- Efficient data transformation
- Optimized rendering
- Smooth animations
- Responsive design

Future Improvements

1. Additional customization options
2. Enhanced mobile responsiveness
3. More interactive features
4. Additional data visualization options

Usage Example

```
import SunburstChart from './components/SunburstChart';

const data = {
  name: "Root",
  value: 100,
  children: [
    {
      name: "Category 1",
      value: 60,
      children: [...]
    },
    {
      name: "Category 2",
      value: 40,
      children: [...]
    }
  ]
};

function App() {
  return <SunburstChart data={data} />;
}
```

Conclusion

This implementation provides a robust and interactive sunburst chart visualization that meets the requirements of displaying hierarchical data in an intuitive and user-friendly manner. The solution combines modern web technologies with best practices in data visualization to create an engaging user experience.