Portfolio Project - Detailed Concepts Documentation

1. React Components

React components are reusable, independent pieces of UI. They can be functional or class-based. In this portfolio, we mostly use functional components with JSX.

```
// Example Functional Component function Header() { return My Portfolio; } export
default Header;
```

2. Props in React

Props (short for properties) are used to pass data from parent to child components. They are read-only and help make components reusable.

```
function Greeting({ name }) { return Hello, {name}!; } // Usage
```

3. State in React (useState Hook)

State is used to store data in a component. The useState hook allows us to manage local component state.

```
import { useState } from 'react'; function Counter() { const [count, setCount] =
useState(0); return ( {count} setCount(count + 1)}>Increase ); }
```

4. React Router

React Router is used for navigation between pages in a React app. It allows creating single-page applications with multiple views.

```
import { BrowserRouter, Route, Routes } from 'react-router-dom'; import Home from
'./pages/Home'; import NotFound from './pages/NotFound'; function App() { return ( }
/> } /> ); }
```

5. Custom Hooks

Custom hooks are reusable functions that use React hooks. In this project, use-toast.js is a custom hook for showing notifications.

```
import { useState } from 'react'; export function useToast() { const [message,
setMessage] = useState(''); return { message, setMessage }; }
```

6. CSS & Styling

Styling in React can be done using CSS files, CSS modules, or styled-components. In this project, plain CSS files are imported into components.

```
import './StarBackground.css'; function Background() { return ; }
```

7. Vite

Vite is a build tool that provides fast development and optimized production builds. It replaces Create React App for better performance.

```
// Run dev server npm run dev // Build for production npm run build
```

8. Toast Notifications

Toast notifications provide feedback messages to users. In this project, toaster.jsx and toast.jsx handle showing notifications.

```
export function Toast({ message }) { return {message}; }
```

9. Responsive Design

Responsive design ensures the portfolio looks good on all screen sizes. CSS media queries are used for responsiveness.

```
@media (max-width: 768px) { .container { flex-direction: column; } }
```