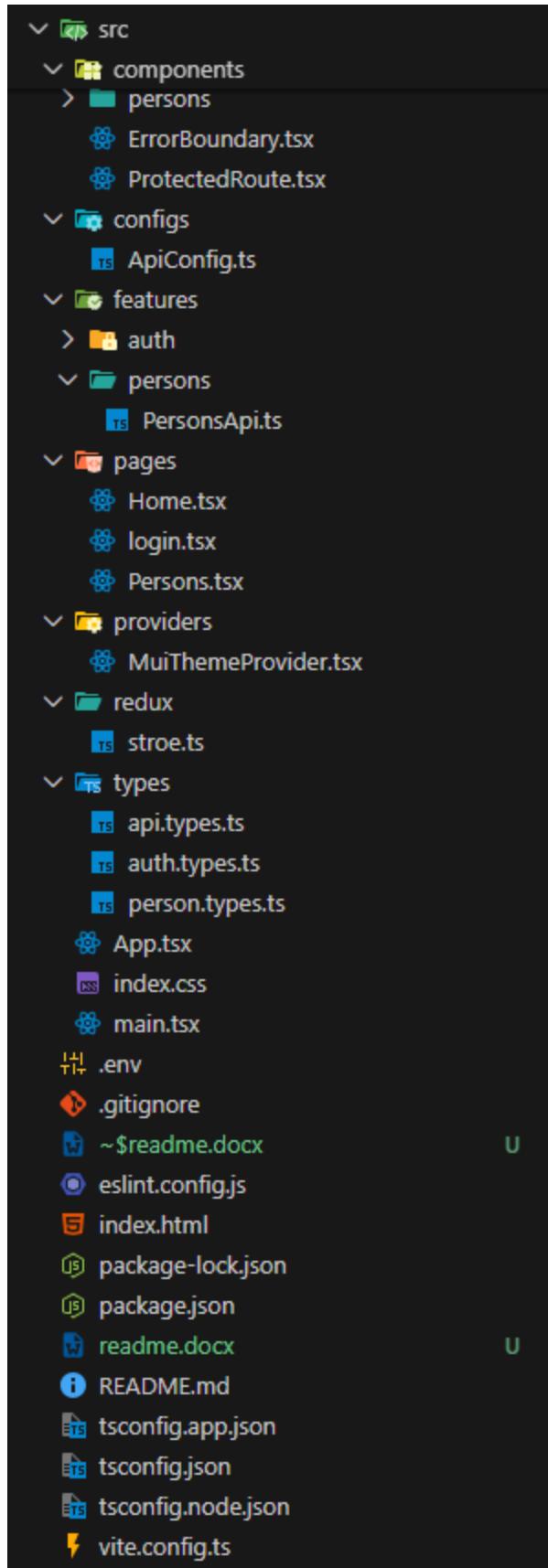


عنوان پروژه : تست مصاحبه فنی شرکت نوآروان

وابستگی های مورد استفاده در پروژه :

```
{  
  "name": "ramin-majidi",  
  "private": true,  
  "version": "0.0.0",  
  "type": "module",  
  "scripts": {  
    "dev": "vite",  
    "build": "tsc -b && vite build",  
    "lint": "eslint .",  
    "preview": "vite preview"  
  },  
  "dependencies": {  
    "@emotion/react": "^11.14.0",  
    "@emotion/styled": "^11.14.1",  
    "@fontsource/roboto": "^5.2.9",  
    "@hookform/resolvers": "^5.2.2",  
    "@mui/icons-material": "^7.3.6",  
    "@mui/material": "^7.3.6",  
    "@mui/stylis-plugin-ttl": "^7.3.6",  
    "@mui/x-date-pickers": "^8.21.0",  
    "@reduxjs/toolkit": "^2.11.1",  
    "date-fns": "^4.1.0",  
    "dayjs": "^1.11.19",  
    "react": "^19.2.0",  
    "react-dom": "^19.2.0",  
    "react-hook-form": "^7.68.0",  
    "react-redux": "^9.2.0",  
    "react-router-dom": "^7.10.1",  
    "stylis": "^4.3.6",  
    "sweetalert2": "^11.26.4",  
    "yup": "^1.7.1"  
  },  
  "devDependencies": {  
    "@eslint/js": "^9.39.1",  
    "@types/node": "^24.10.1",  
    "@types/react": "^19.2.5",  
    "@types/react-dom": "^19.2.3",  
    "@types/react-redux": "^7.1.34",  
    "@types/stylis": "^4.2.7",  
    "@vitejs/plugin-react": "^5.1.1",  
    "eslint": "^9.39.1",  
    "eslint-plugin-react-hooks": "^7.0.1",  
    "eslint-plugin-react-refresh": "^0.4.24",  
    "globals": "^16.5.0",  
    "typescript": "~5.9.3",  
    "typescript-eslint": "^8.46.4",  
    "vite": "^7.2.4"  
  }  
}
```



ساختار کلی پروژه و چیدمان فایل ها : برای ساختار کامپوننت ها تلاش شده از سیستم اتمیک دیزاین و فیچر بیس بودن کامپوننت ها پیروی و استفاده شود .

از جمله کامپونت های ساخته شده برای هندل کردن خطاهای احتمالی در پروژه و جلوگیری از کرش کردن کلی برنامه و اینکه کل درخت کامپونت ها در صورت بروز خطای دچار مشکل نشوند و برنامه به روند خود ادامه بدهد به کامپونت ErrorBoundary میتوان اشاره کرد.

```
import { Component, type ErrorInfo, type ReactNode } from "react";
interface Props {
  children: ReactNode;
  fallback?: ReactNode;
}
interface State {
  hasError: boolean;
  error?: Error;
}
class ErrorBoundary extends Component<Props, State> {
  constructor(props: Props) {
    super(props);
    this.state = { hasError: false };
  }

  static getDerivedStateFromError(error: Error): State {
    return { hasError: true, error };
  }

  componentDidCatch(error: Error, errorInfo: ErrorInfo) {
    console.error("ErrorBoundary caught an error:", error, errorInfo);
  }

  render() {
    if (this.state.hasError) {
      if (this.props.fallback) {
        return this.props.fallback;
      }

      return (
        <div style={{ padding: "20px", textAlign: "center" }}>
          <h3>An error has occurred.</h3>
          <p>Please refresh the page or contact support.</p>
          {import.meta.env.NODE_ENV === "development" && (
            <details style={{ marginTop: "10px", textAlign: "left" }}>
              <summary>Error details</summary>
              <pre>{this.state.error?.toString()}</pre>
            </details>
          )}
        </div>
      );
    }

    return this.props.children;
  }
}

export default ErrorBoundary;
```

همچنین کامپوننت ProtectedRoute که از استاندارد HOC در React پیروی میکند که برای هندل کردن روت های محافظت شده در پروژه استفاده میشود.

```
import React from "react";
import { Navigate, useLocation } from "react-router-dom";
import { useSelector } from "react-redux";
import { selectIsAuthenticated } from "../features/auth/authSlice";

interface ProtectedRouteProps {
  children: React.ReactNode;
  requireAuth?: boolean;
  redirectTo?: string;
}

const ProtectedRoute: React.FC<ProtectedRouteProps> = ({  
  children,  
  requireAuth = true,  
  redirectTo = "/login",  
}) => {  
  const isAuthenticated = useSelector(selectIsAuthenticated);  
  const location = useLocation();  
  
  if (requireAuth && !isAuthenticated) {  
    // Redirect به login page با حفظ location  
    return <Navigate to={redirectTo} state={{ from: location }} replace />;  
  }  
  
  if (!requireAuth && isAuthenticated) {  
    // اگر authenticated بشدت و requireAuth نباشد، به homepage redirect  
    return <Navigate to="/persons" replace />;  
  }  
  
  return <>{children}</>;  
};  
  
export default ProtectedRoute;
```

