

Entwicklerdokumentation

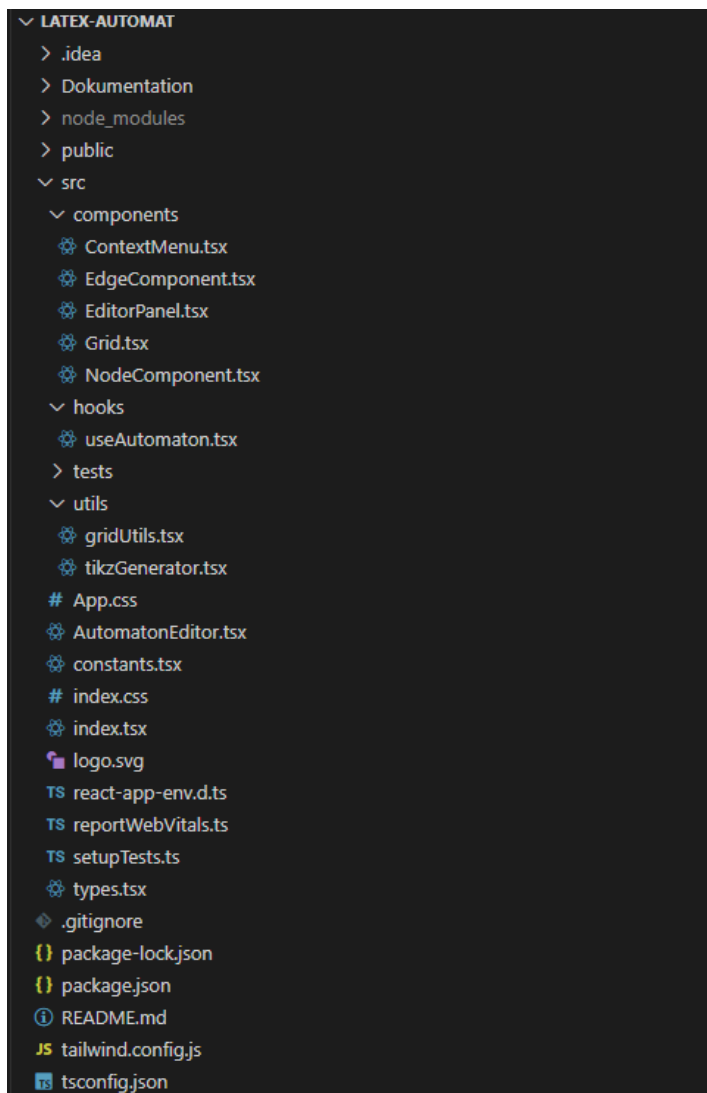
Projektübersicht:

- Name: LaTeX-Automaten
- Beschreibung: Eine React-Anwendung, mit welcher man Automaten erstellen kann. Zusätzlich wird zu den Automaten automatisch TikZ-Code für LaTeX Export generiert.

Architektur:

- Frontend: React mit Typescript
- Backend: Node.js
- Styling: Tailwind CSS
- Build/Run: Create React App
- Tests: Jest

Projektstruktur:



```

└─ LATEX-AUTOMAT
  ├── .idea
  ├── Dokumentation
  ├── node_modules
  ├── public
  └── src
      ├── components
      │   ├── ContextMenu.tsx
      │   ├── EdgeComponent.tsx
      │   ├── EditorPanel.tsx
      │   ├── Grid.tsx
      │   └── NodeComponent.tsx
      ├── hooks
      │   └── useAutomaton.tsx
      ├── tests
      ├── utils
      │   ├── gridUtils.tsx
      │   └── tikzGenerator.tsx
      ├── # App.css
      ├── AutomatonEditor.tsx
      ├── constants.tsx
      ├── # index.css
      ├── index.tsx
      ├── logo.svg
      ├── TS react-app-env.d.ts
      ├── TS reportWebVitals.ts
      ├── TS setupTests.ts
      ├── types.tsx
      ├── .gitignore
      ├── {} package-lock.json
      ├── {} package.json
      ├── README.md
      ├── JS tailwind.config.js
      └── TS tsconfig.json

```

Wichtige Elemente:

ContextMenu.tsx → Stellt ein Kontextmenü dar, über das verschiedene Aktionen ausgeführt werden können

EdgeComponent.tsx → Darstellung einer Kante

EditorPanel.tsx → Panel zur Bearbeitung von Knoten- und Kanteneigenschaften, z. B. Beschriftung der Kanten/Knoten

Grid.tsx → Darstellung eines Rasters

NodeComponent.tsx → Darstellung eines Knotens inklusive Start- und Endzustand

useAutomaton.tsx → Enthält Logik für das State-Management des Automaten

gridUtils.tsx → Sammlung von Hilfsfunktionen für die Arbeit mit dem Grid

tikzGenerator.tsx → Enthält Logik zur Umwandlung des Automaten in TikZ-Code

AutomatonEditor.tsx → Hauptelement für die Bearbeitung des Automaten. Bindet die anderen Komponenten zusammen und steuert den Gesamtworkflow