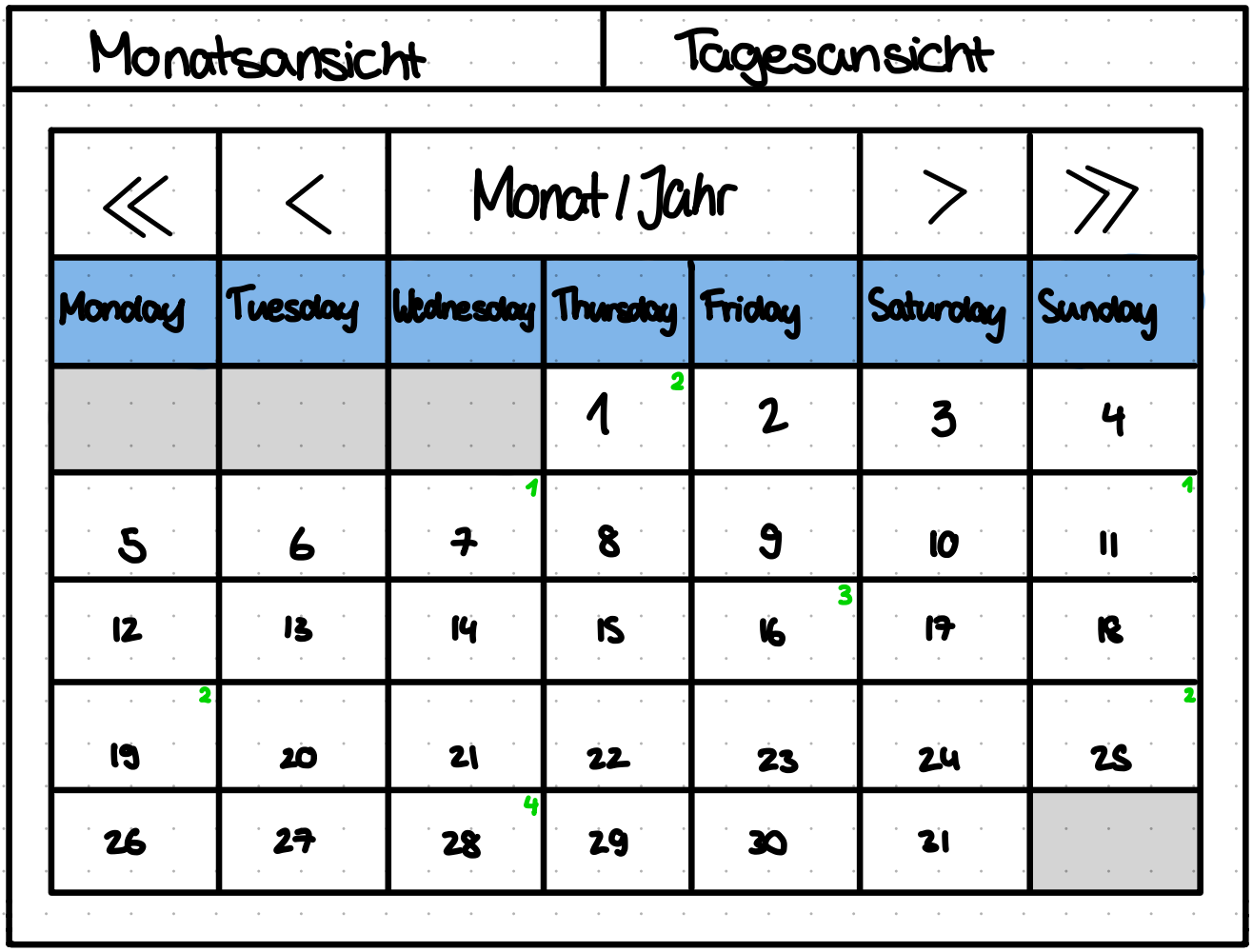
Matrikelnummer: 8944534

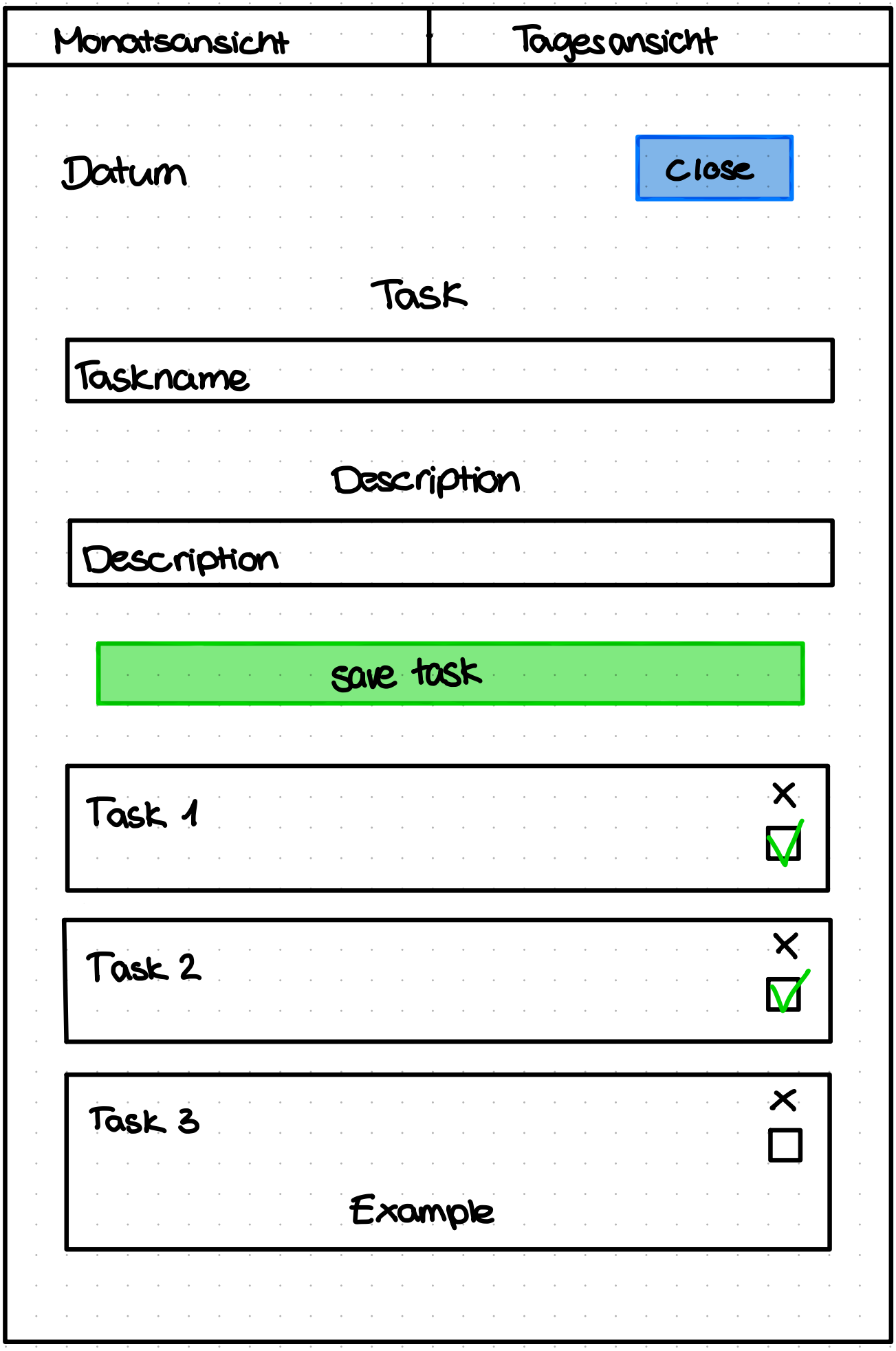
Kommentare in month.jsx und Cells.jsx wurden aus irgendeinem Grund in Github nicht hochgeladen, sind aber in der Zip auf Moodle verfügbar.

Paper Prototype

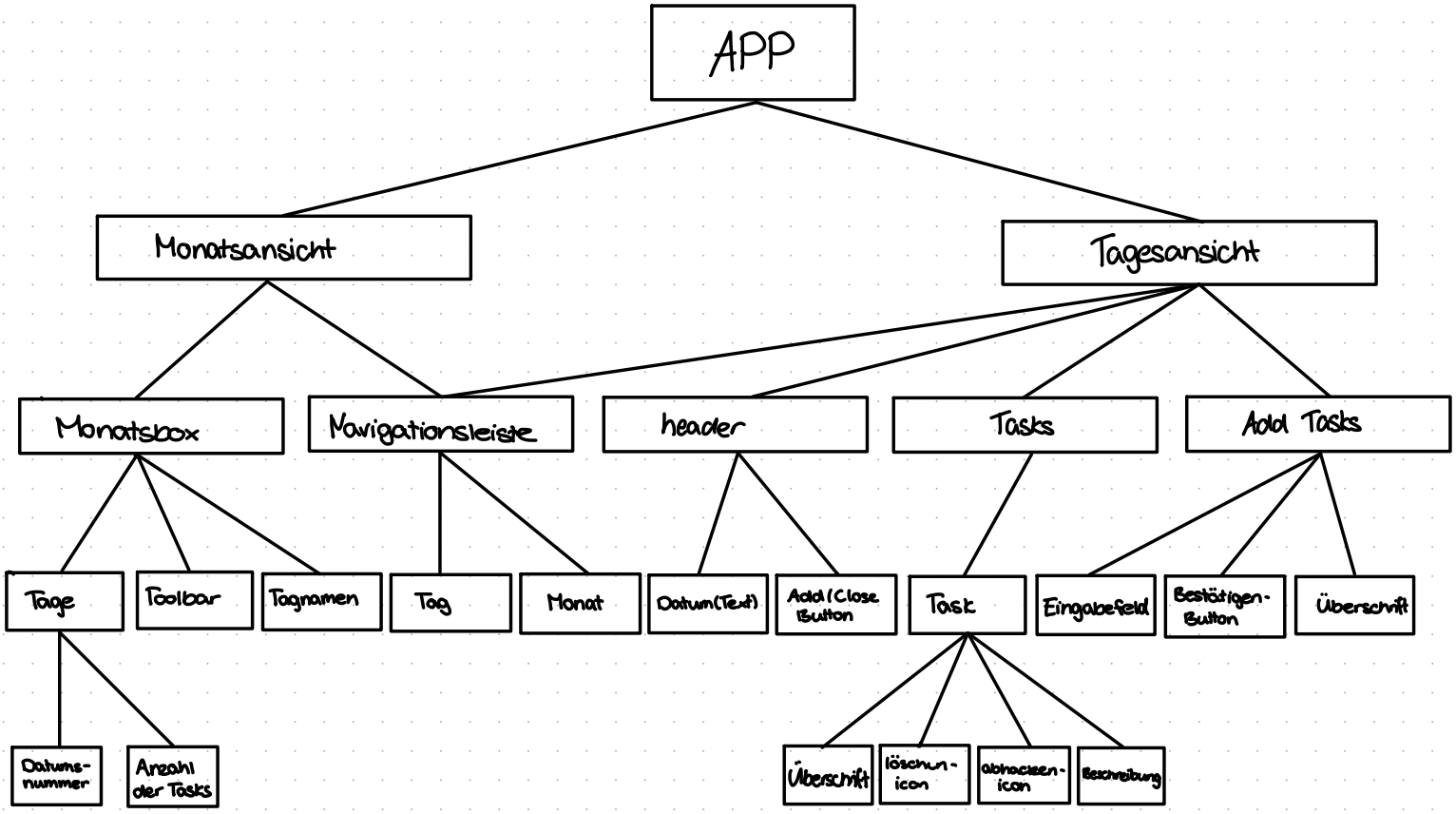
Monatsansicht



Tagesansicht



Klassendiagramm



Requirements:

Eine To-Do-Liste ist eine nützliche App, die es ermöglicht, Aufgaben und Aktivitäten zu organisieren und zu verwalten. Um eine solche App zu erstellen, sind verschiedene Anforderungen zu berücksichtigen.

Zunächst einmal sollte die App mindestens aus zwei Seiten bestehen: einer Startseite und einer Tagesansicht. Auf der Startseite soll ein Kalender in einer Monatsansicht angezeigt werden, bei welchem die Anzahl der ToDos eines Tages im Feld des jeweiligen Tages stehen soll. So kann der Benutzer schnell erkennen, an welchen Tagen besonders viele Aufgaben anstehen. Durch einen Klick auf einen Tag soll sich die Tagesansicht öffnen, auf welcher alle ToDos des Tages sichtbar sind. Außerdem können dort neue ToDos erstellt, gelöscht oder abgehakt werden.

Eine Navigationsleiste sollte dabei helfen, zwischen der Monatsansicht und der Tagesansicht zu wechseln. Diese Navigationsleiste sollte intuitiv und einfach zu bedienen sein, damit der Benutzer schnell zwischen den verschiedenen Ansichten wechseln kann.

Neben den funktionalen Anforderungen sollte auch auf nicht-funktionale Anforderungen geachtet werden. Die Benutzerfreundlichkeit ist hierbei besonders wichtig, da die App einfach und intuitiv zu bedienen sein sollte.

Konzept:

Um die ToDoListe App mit React zu implementieren, könnte die Architektur aus mehreren Komponenten bestehen. Eine mögliche Umsetzung könnte folgendermaßen aussehen:

App: Die zentrale Komponente, die alle anderen Komponenten umfasst und den Zustand der App verwaltet.
MonthView: Eine Komponente, die die Monatsansicht mit dem Kalender und der Anzahl der ToDos pro Tag darstellt. Diese Komponente könnte verschiedene Unter-Komponenten wie Day, MonthHeader und Navigation enthalten.
DayView: Eine Komponente, die die Tagesansicht mit den ToDos eines bestimmten Tages darstellt. Diese Komponente könnte verschiedene Unter-Komponenten wie ToDo, ToDoForm und Navigation enthalten.
Two-Way-Binding könnte genutzt werden, um die Auswahl der Tage in der Monatsansicht zu gestalten. Dazu könnte eine stateful Komponente, z.B. MonthView, den ausgewählten Tag als State halten. Sobald der Benutzer auf einen Tag klickt, wird der entsprechende Tag als aktiver Tag im State gespeichert und die Komponente wird erneut gerendert. Die DayView Komponente würde dann basierend auf dem ausgewählten Tag gerendert werden.

Routing könnte genutzt werden, um zwischen der Monatsansicht und der Tagesansicht zu navigieren. Dazu könnte React Router eingesetzt werden. Wenn der Benutzer auf einen Tag in der Monatsansicht klickt, könnte er automatisch zur Tagesansicht für den ausgewählten Tag navigieren. Umgekehrt könnte es eine Zurück-Schaltfläche in der Tagesansicht geben, die den Benutzer zur Monatsansicht zurückbringt.

Um den Zustand der App zu verwalten, könnte der Zustand in der App-Komponente gehalten werden. Der Zustand könnte Informationen wie die ausgewählten Termine, die ToDos und deren Status enthalten. Conditional Rendering könnte genutzt werden, um nur die relevanten Komponenten basierend auf dem ausgewählten Tag und dem Zustand der ToDos anzuzeigen.

Listrendering könnte ebenfalls genutzt werden, um die ToDos in der Tagesansicht darzustellen. Eine ToDo-Liste könnte als Array im State gespeichert werden und die ToDo-Komponente könnte für jedes Element in der Liste gerendert werden. Das Löschen oder Abhaken eines ToDo-Elements würde den Zustand der ToDo-Liste im State aktualisieren und ein erneutes Rendern der Liste auslösen.

Insgesamt wäre die ToDoListe App eine React-Anwendung mit einer klaren Architektur, die React-Elemente wie Two-Way-Binding, Routing, Conditional Rendering und Listrendering nutzt, um eine benutzerfreundliche ToDoListe zu erstellen.

Resümee:

Trotz des Umstands, dass ich mitunter aufgrund einer Krankheit nicht in der Lage war, das Projekt vollständig abzuschließen, habe ich dennoch wertvolle Erfahrungen in der Implementierung einer ToDoListe App mit React gesammelt. Ich konnte erfolgreich die Monats- und Tagesansicht in separate Komponenten aufteilen und diese mit ihren jeweiligen Unter-Komponenten umsetzen. Das Two-Way-Binding ist mir gut gelungen und ich konnte den Zustand der Monatsansicht basierend auf der Auswahl des Benutzers aktualisieren. Allerdings war es mir nicht möglich, das Routing in die App zu integrieren, um zwischen den beiden Ansichten zu navigieren. Aus diesem Grund besteht die App nun nur aus beiden Seiten, die direkt untereinander auf einer Seite eingefügt sind. Insgesamt war es eine Herausforderung, aber ich habe wertvolle Erfahrungen in der Arbeit mit React gesammelt und bin motiviert, meine Kenntnisse weiter auszubauen.