

Bachelorarbeit

Konzeptionierung und Implementierung einer Pivot-Tabellen-Komponente in einem CRM-System

zur Erlangung des akademischen Grades

Bachelor of Science (B. Sc.)

vorgelegt dem

Fachbereich Mathematik, Naturwissenschaften und Informatik
der Technischen Hochschule Mittelhessen

vorgelegt von

Marcel Frank Kucera

im September 2025

Referent: Sebastian Süß, M.Sc.

Korreferent: Prof. Dr. Steffen Vaupel

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Zielsetzung	2
1.3	Methodik	3
2	Kontext	4
2.1	Pivot-Tabellen	4
2.2	CURSOR-CRM	5
2.3	Infoboards und Kacheln im CURSOR-CRM	6
2.3.1	Kachel V1 zu V2	6
2.4	React	7
3	Konzeptionierung	9
3.1	Betrachtung alte Pivot-Tabellen-Komponente	9
3.2	Ziele der neuen Implementierung	9
3.3	Anforderungen	9
3.4	Design	9
4	Technischer Entwurf	11
4.1	Architektur	11
4.2	datenverarbeitung	11
4.3	Evaluation und Auswahl von verfügbaren Bibliotheken	11
5	Implementierung	12
5.1	Freistehende Entwicklung	12
5.2	Integration in das CRM-System	12
5.3	Vergleich Tabellenkomponente MUI-X zu Eigenentwicklung	12
5.4	Tests	12
6	Evaluation	13
6.1	Usability-Tests	13
6.2	Bewertung	13
7	Fazit und Ausblick	14
	Bibliographie	15

TODO: Eidstattliche erklärung

TODO: KI Erklärung

TODO: Zusammenfassung

1 Einleitung

- teils von exposee übernehmen

1.1 Motivation

In der heutigen datengetriebenen Geschäftswelt spielen Customer Relationship Management (CRM)-Systeme eine zentrale Rolle bei der Kundengewinnung, -bindung sowie beim Aufbau und der Pflege langfristiger Kundenbeziehungen [1, S. 342]. CRM-Systeme erfassen und verarbeiten dabei eine Vielzahl von Informationen – von Kundenkontakten und Interaktionen über Verkaufschancen bis hin zu Serviceanfragen und -tickets.

Doch die bloße Verfügbarkeit großer Datenmengen führt nicht automatisch zu besseren Entscheidungen. Ohne benutzerfreundliche und leistungsfähige Analysetools bleibt das Potenzial dieser Daten weitgehend ungenutzt. Besonders Vertriebs- und Marketingteams stehen regelmäßig vor analytischen Fragestellungen wie: „In welchen Regionen sind unsere Produkte im letzten Jahr unterdurchschnittlich gelaufen?“ oder „Wie hat sich der Umsatz nach Vertriebsregionen in den letzten drei Jahren entwickelt?“ Solche Fragen zielen darauf ab, Entwicklungen nachzuvollziehen und aufkommende Trends frühzeitig zu erkennen, um fundierte strategische Entscheidungen treffen zu können [1, S. 345].

Pivot-Tabellen sind ein bewährtes Mittel, um genau solche Fragestellungen interaktiv und effizient zu beantworten. Sie ermöglichen es den Nutzern, komplexe Datenbestände flexibel und intuitiv zu gruppieren, zu aggregieren und auszuwerten. Beispielsweise können Verkaufszahlen nach Vertriebsregion, Jahr oder Kundensegment gegliedert und analysiert werden. Dabei lassen sich zentrale Kennzahlen wie Umsätze oder Stückzahlen mithilfe von verschiedenen Aggregationsfunktionen (Summe, Durchschnitt, etc.) berechnen und übersichtlich in Tabellenform darstellen. Ergänzend dazu bieten weitere mögliche Visualisierungsmöglichkeiten der Pivot-Tabelle wie Balken-, Linien- oder Kreisdiagramme eine anschauliche Aufbereitung der Ergebnisse, wodurch sich Muster und Entwicklungen schneller erfassen und kommunizieren lassen.

Die Implementierung einer Pivot-Komponente direkt im CRM-System bietet aus technischer und organisatorischer Sicht mehrere Vorteile. Insbesondere ermöglicht sie die Nutzung der Datensätze an zentraler Stelle im System, wo die Daten bereits erfasst und

gepflegt werden. Dadurch entfällt die Notwendigkeit, die Daten in externe Anwendungen zu exportieren und dort weiterzuverarbeiten, was mit zusätzlichem Aufwand und potenziellen Inkonsistenzen verbunden sein kann.

Darüber hinaus erleichtert die Integration einer solchen Komponente die Einbettung analytischer Methoden in bestehende Arbeitsabläufe und Prozesse. Anwender können direkt innerhalb der vertrauten CRM-Systemumgebung Auswertungen interaktiv erstellen und nutzen, ohne zwischen verschiedener Software wechseln zu müssen. Auf diese Weise kann die Auswertung vorhandener Daten, sowie die darauf aufbauende Entscheidungsfindung, unterstützt werden.

1.2 Zielsetzung

Das Ziel dieser Bachelorarbeit ist die Konzeption und Umsetzung einer Pivot-Tabellen-Komponente im Kontext des CURSOR-CRM-Systems (<https://www.cursor.de>). Es stellt über sogenannte Infoboards eine Oberfläche bereit, auf der verschiedene Kacheln platziert werden können. Diese Kacheln ermöglichen unter anderem Datenvisualisierungen und -analysen. Die Komponenten lassen sich flexibel in das Infoboard einfügen und können dabei auf den Kontext der jeweiligen Umgebung als Datenquelle zurückgreifen.

Die entwickelte Komponente soll als eine „Infoboard-Kachel“ in die bestehende Systemarchitektur integriert werden und somit Zugriff auf die vorhandenen Datenquellen, wie Suchen, Masken und Unterbereiche, haben, sowie eine einheitliche Oberfläche zur Konfiguration der Komponente anbieten.

Im System existiert bereits eine Pivot-Tabellen-Komponente für Infoboards. Diese basiert jedoch auf veralteten Technologien, die im kommenden Jahr aus dem System entfernt werden sollen. Darüber hinaus hat die derzeitige Version nur begrenzte Anpassungsmöglichkeiten an kundenspezifische Anforderungen. Aus diesen Gründen ergibt sich der Bedarf für die Entwicklung einer aktualisierten, flexibler einsetzbaren und zukunftssicheren Version.

Die Integration in bestehende Arbeitsprozesse sowie CRM-Funktionalitäten stehen bei der Umsetzung im Vordergrund. Die Komponente soll so gestaltet sein, dass sie für den Nutzer einfach konfigurierbar und intuitiv bedienbar ist. Zudem soll die Komponente performant sein, um auch bei großen Datenmengen eine flüssige Benutzererfahrung zu bieten. Des Weiteren soll die entwickelte Komponente die firmeninternen Entwick-

lungsstandards einhalten sowie bestehenden Code wiederverwenden, damit sie vom Entwicklungsteam auch in Zukunft einfach gewartet werden kann.

1.3 Methodik

Die Arbeit wird in mehrere aufeinanderfolgende Phasen gegliedert.

Zu Beginn erfolgt eine Anforderungsanalyse, auf deren Grundlage die Konzeption der Pivot-Komponente entwickelt wird. Im Anschluss daran wird ein technischer Entwurf erstellt, bei dem auch geeignete externe Bibliotheken hinsichtlich ihrer Eignung und Kompatibilität untersucht werden.

Die Implementierung der Komponente erfolgt iterativ. In einer ersten Phase wird ein unabhängiger Prototyp entwickelt, der zunächst losgelöst vom CURSOR-CRM-System funktioniert. Nach Fertigstellung eines funktionsfähigen Prototyps erfolgt die Integration in die CRM-Umgebung, einschließlich der Implementierung erforderlicher Schnittstellen. Dabei werden bestehende Systemlogiken und Bibliotheken, soweit möglich, übernommen und wiederverwendet.

Parallel zur technischen Entwicklung werden fortlaufend Usability-Tests durchgeführt, um frühzeitig Rückmeldungen zur Benutzerfreundlichkeit zu erhalten und diese in den Entwicklungsprozess einfließen zu lassen.

2 Kontext

2.1 Pivot-Tabellen

Pivot-Tabellen sind ein hilfreiches Werkzeug zur Datenanalyse und zum Reporting. Häufig werden sie im Zusammenhang mit Tabellenkalkulationssoftware wie Microsoft Excel verwendet, wo sie es ermöglichen, große Datenmengen interaktiv zu gruppieren, zu filtern, zu aggregieren und zu visualisieren.

Das Grundprinzip besteht darin, Datenfelder in Zeilen- und Spaltenachsen anzuordnen. Dabei werden numerische Werte aus einem anderem Feld der jeweiligen Daten an den Kreuzungen aggregiert, beispielsweise als Summe oder Durchschnitt. Über Filterfunktionen lassen sich spezifische Ausschnitte der Daten betrachten. So kann beispielsweise ausgewertet werden, wie viele Produkte ein bestimmter Vertriebsmitarbeiter in einem Monat verkauft hat.

Moderne Pivot-Tabellen ergänzen die tabellarische Ansicht häufig durch Diagramme wie Balken- oder Kreisdiagramme, um Muster und Trends besser sichtbar zu machen. Insgesamt abstrahieren Pivot-Tabellen SQL-ähnliche Operationen in einer benutzerfreundlichen Oberfläche, sodass auch fachliche Anwender komplexe Datenanalysen ohne Programmier- oder Datenbankkenntnisse durchführen können.

	Monat	April	Januar	Juli	Oktober	Totals
	Verkäufer					
Dieter		220.00	159.00	230.00	215.00	824.00
Julia		230.00	200.00	270.00	240.00	940.00
Sabine		210.00	180.00	235.00	250.00	875.00
Thomas		180.00	140.00	210.00	170.00	700.00
Totals		840.00	679.00	945.00	875.00	3,339.00

Abbildung 1: Beispiel einer Pivottabelle mit Summierung der Auftragswerte [2]

2.2 CURSOR-CRM

Das CURSOR-CRM ist ein Customer Relationship Management System, das Unternehmen bei der strukturierten Erfassung, Verwaltung und Auswertung ihrer Kunden- und Geschäftsdaten unterstützt. Eine zentrale Funktion des Systems besteht in der Speicherung verschiedenartiger Daten, welche Entitäten genannt werden.

Entitäten beschreiben eine Klasse gleichartiger Datenobjekte. Sie definieren die Felder und Eigenschaften der jeweiligen Datensätze, wie zum Beispiel Geschäftspartner, Mitarbeitende, Angebote oder Aktivitäten. Jeder Datensatz gehört zu einer bestimmten Entität und kann mit anderen Datensätzen verknüpft sein. Die Anzeige und Bearbeitung der einzelnen Datensätze erfolgt über sogenannte Masken, in denen die zugehörigen Felder benutzerfreundlich dargestellt werden. Diese Masken sind anpassbar und lassen sich durch Scripting erweitern, um individuelle Anforderungen und Prozesse abzubilden.

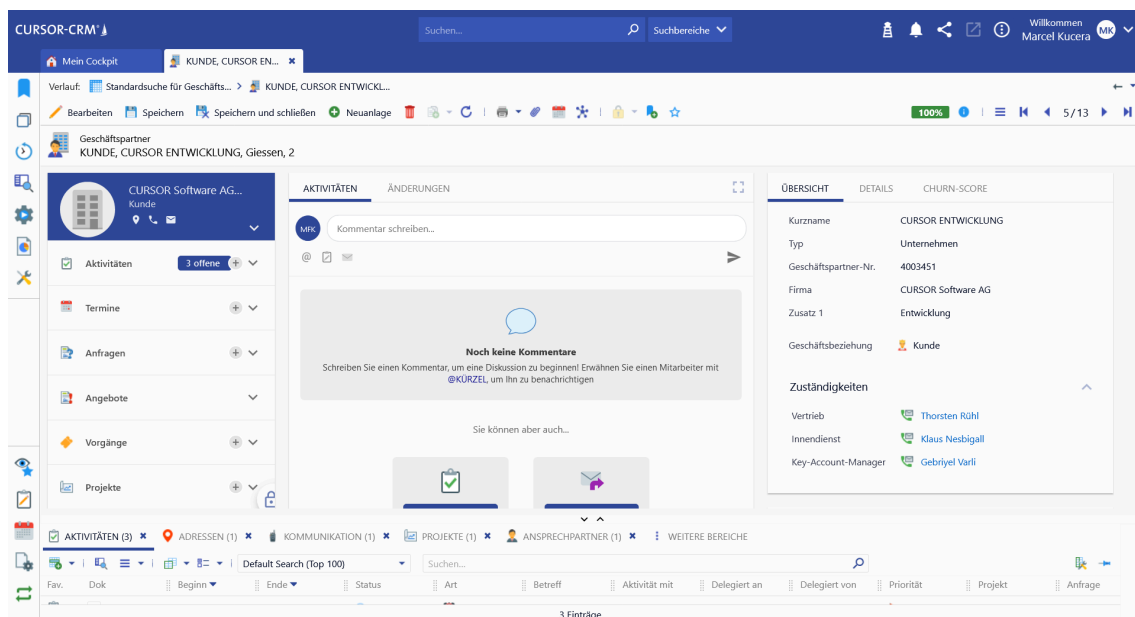


Abbildung 2: Screenshot des CURSOR-CRM auf der Maske für einen Geschäftspartner

Das CURSOR-CRM lässt sich flexibel an individuelle Unternehmensanforderungen anpassen. Neue Entitäten können hinzugefügt und bestehende modifiziert werden. Über vielfältige Schnittstellen ist zudem eine Integration mit externen Systemen möglich, sodass ein reibungsloser Datenaustausch innerhalb bestehender IT-Landschaften gewährleistet ist.

Im CURSOR-CRM stehen sogenannte „Suchen“ zur Verfügung, mit denen Anwender gezielt Daten anhand bestimmter Kriterien auffinden können. Diese Suchen sind ein zentraler Bestandteil des Systems und ermöglichen es, nicht nur innerhalb einzelner Entitäten zu filtern, sondern auch über deren Beziehungen hinweg komplexe Datenabfragen durchzuführen. Nutzerinnen und Nutzer haben dabei die Möglichkeit, eigene Suchen zu erstellen und individuell anzupassen. Zudem lässt sich festlegen, welche Felder in den Suchergebnissen angezeigt werden, sodass die Darstellung exakt auf die jeweiligen Anforderungen abgestimmt werden kann.

2.3 Infoboards und Kacheln im CURSOR-CRM

Infoboards im CURSOR CRM dienen als flexible Oberfläche zur Visualisierung und Strukturierung von Informationen. Sie können in unterschiedlichen Kontexten eingesetzt werden, etwa in Dashboards, innerhalb von Masken oder den Unterbereichen der Masken, wo die verknüpften Entitäten angezeigt werden. Auf diesen Infoboards lassen sich sogenannte Kacheln platzieren, die individuell konfigurierbar sind und unterschiedliche Funktionen übernehmen können.

Kacheln ermöglichen es, kontextbezogene Daten dynamisch zu verarbeiten und visuell aufzubereiten. Dabei greifen sie auf Informationen aus dem jeweiligen Anzeigekontext zu, verarbeiten diese und stellen sie in geeigneter Form für den Nutzer dar. Typische Anwendungsbeispiele sind ToDo-Listen, die offene Aktivitäten anzeigen, grafische Darstellungen wie Diagramme oder einfache tabellarische Auflistungen relevanter Datensätze. Die Anordnung der Kacheln auf einem Infoboard kann per Drag-and-Drop angepasst werden, wodurch sich individuelle Arbeitsoberflächen schnell und intuitiv gestalten lassen. Kacheln lassen sich über ein Konfigurationsmenü individuell anpassen, wobei neben allgemeinen Einstellungen wie Inhalt, Darstellung und Sortierung auch kachelspezifische Optionen zur Verfügung stehen.

2.3.1 Kachel V1 zu V2

Im aktuellen System existieren zwei Versionen des Kachel-Systems, die sich hinsichtlich ihrer technologischen Grundlage unterscheiden. Die erste Version, welche als Kachel V1 bezeichnet wird, basiert auf klassischen Webtechnologien wie Vanilla JavaScript, HTML und CSS. Diese Variante ist funktional weiterhin im Einsatz, soll jedoch perspektivisch vollkommen durch die modernere Lösung Kachel V2 ersetzt werden.

Das neue Kachel V2 System basiert auf dem JavaScript-Framework React. Durch die Verwendung von React ist es möglich, auf eine bestehende Sammlung von wiederverwendbaren Komponenten innerhalb des CURSOR-CRM zuzugreifen. Dies ermöglicht ein konsistenteres Erscheinungsbild zwischen der Systemoberfläche und den Kacheln und kann die Entwicklung und Wartung neuer Kacheln erleichtern. Darüber hinaus bietet diese zweite Version eine verbesserte Wart- und Erweiterbarkeit, da sie auf modernen Entwicklungsstandards aufbaut.

TODO: in erfahrung bringen warum genau gewechselt wird

2.4 React

React ist eine von Meta entwickelte JavaScript-Bibliothek zur Erstellung dynamischer Benutzeroberflächen. Sie basiert auf einer komponentenbasierten Architektur, bei der wiederverwendbare UI-Elemente, sogenannte Komponenten, sowohl Struktur als auch Verhalten kapseln. Dies ermöglicht eine modulare und übersichtliche Entwicklung komplexer Webanwendungen.

Ein zentrales Konzept von React ist der Virtual DOM, eine abstrahierte Repräsentation des tatsächlichen DOM im Speicher. Änderungen am Anwendungszustand werden dort simuliert und durch einen Vergleichsalgorithmus gezielt und effizient auf den realen DOM übertragen. Dadurch wird die Performance bei der Darstellung dynamischer Inhalte deutlich verbessert.

React folgt einer deklarativen Programmierweise. Entwicklerinnen und Entwickler beschreiben den gewünschten UI-Zustand, während React sich um dessen Umsetzung kümmert. Zum Einsatz kommt dabei JSX, eine Syntaxerweiterung, die HTML-ähnliche Strukturen direkt in JavaScript integriert. Das erleichtert die Strukturierung und Wartung von Komponenten.

React ist als Bibliothek konzipiert und konzentriert sich hauptsächlich auf die Darstellung von Benutzeroberflächen. Durch diesen modularen Aufbau lässt sich React problemlos in bestehende Anwendungen integrieren, beispielsweise um einzelne UI-Komponenten schrittweise zu modernisieren oder interaktiver zu gestalten. Gleichzeitig besteht jedoch auch die Möglichkeit, mit Hilfe von darauf aufbauenden Frameworks komplette Anwendungen umzusetzen.

TODO: typescript?

3 Konzeptionierung

3.1 Betrachtung alte Pivot-Tabellen-Komponente

- eine der komplexesten kacheln
- grundlegende funktionen. analyse welche davon übernommen werden sollen?
- veraltete technologien
 - historie v1?
- umständliche konfiguration
- hinterliegende bibliothek wird nichtmehr gewartet (<https://github.com/nicolaskrucht/en/pivottable>)

3.2 Ziele der neuen Implementierung

- modernere technologien
- nutzerfreundlichkeit
- interne wartbarkeit
- verwendung von internen komponenten
- einheitliche oberfläche
- excelexport?
- hier nochmal auf den aspekt der einfachen datenanalyse eingehen (insbesondere zeigen, dass das wirklich wert hat)

3.3 Anforderungen

- betrachtung der stakeholder (kunde, produktmanagement, entwickler, (berater?))
- zusammen mit dem product management erarbeitet
- woran messe ich die qualität?
- abnahmekriterien
- umfrage?
- die typischen anwendungsfälle
 - userstories?
 - konfiguration

3.4 Design

- das sollte vielleicht eher in konzeption
- hier vielleicht auf das alte design eingehen und „modernisieren“

- was sind die probleme von dem alten design und wie verbessere ich diese mite dem neuen

4 Technischer Entwurf

4.1 Architektur

- Die Komponente ist in Rahmen und visualisierung aufgeteilt
- abwägen ob die komponente komplett im kacheleditor entwickelt werden soll oder im standard und dann die komponente im kachel kontext verwenden
- rahmen ist komplett selbst
- visualisierungen sind hauptsächlich bibliotheken mit ausnahme von
- diagram zum datenmodell mit client
- auswertung wird auf dem client ausgeführt

4.2 datenverarbeitung

- hier weiß ich nicht so 100%, ob das wirklich benötigt ist, oder ob alle visualisierungs-libraries mir hier die arbeit abnehmen

4.3 Evaluation und Auswahl von verfügbaren Bibliotheken

- Eine Drag and Drop Library wird bereits im System verwendet. Diese wurde für diese Komponente auch verwendet.
- Kurz beschreiben was die macht, dass die eine einfachere version von react-dnd ist, aber für unsere anwendungsfälle reicht
- ein paar pivot libraries raussuchen. dabei eingehen, dass mui-x verwendet wird, aber an sich nicht den anforderungen entspricht
- der react port der bisher verwendeten version wir nichtmehr gewartet und ist mit neueren react versionen inkompatibel
 - hier auch ein bild der fehlermeldung einbinden
- visualisierungslibrary ist auch schon größtenteils vorgegeben (mui charts)
 - adapterkomponenten für die jeweiligen visualisierungen
 - erweiterbarkeit

5 Implementierung

5.1 Freistehende Entwicklung

- prototypen
- einrichtung vite react
- bilder von dem jetzigen stand außerhalb des crm-systems
- entwicklung des rahmens
- entwicklung der klasse für zentrale datenverarbeitung
- entwicklung der tabelle
- entwicklung der adapter

5.2 Integration in das CRM-System

- hier auch genau erläutern wie die kacheln eingebunden werden
 - auch auf die verbindung von system, komponente im standard, bis hin zur kachel gehen
- zuerst eine einbindung in das system
- dann integration in eine kachel
- bei den kacheln auch die kachel api im detail erleutern
- bilder von dem prototypen
- ab hier auch regelmäßig stand checken gegenüber product management und entwicklung

5.3 Vergleich Tabellenkomponente MUI-X zu Eigenentwicklung

- hier nochmal erwähnen, dass die pivot-komponente erst während der entwicklung veröffentlicht wurde

5.4 Tests

6 Evaluation

6.1 Usability-Tests

6.2 Bewertung

7 Fazit und Ausblick

- gut
- erfüllt allen anforderungen

Bibliographie

- [1] W. Becker u. a., Hrsg., *Geschäftsmodelle in der digitalen Welt: Strategien, Prozesse und Praxiserfahrungen*. Wiesbaden: Springer Fachmedien Wiesbaden, 2019. doi: [10.1007/978-3-658-22129-4](https://doi.org/10.1007/978-3-658-22129-4).
- [2] „React-Pivottable“. Zugegriffen: 12. Juli 2025. [Online]. Verfügbar unter: <https://react-pivottable.js.org/>